

QoS em WIMAX – Análise e avaliação de módulos para simulação com o ns-2

Fernando Miguel Ribeiro Bartilotti de Matos



Mestrado em Engenharia Electrotécnica e de Computadores

Área de Especialização de Telecomunicações

Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

2013

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Disciplina de
Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Electrotécnica e de
Computadores

Candidato: Fernando Miguel Ribeiro Bartilotti de Matos, N°1010600

Orientação científica: Prof. Doutor Jorge Botelho Mamede



Mestrado em Engenharia Electrotécnica e de Computadores

Área de Especialização de Telecomunicações

Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

19 de setembro de 2013

Agradecimentos

Um enorme obrigado e as maiores felicidades ao Professor Jorge Mamede, por facultar esta oportunidade de aprendizagem e experiência, por todo o esforço e dedicação em contribuir para o melhor trabalho possível e para que tudo corresse dentro do previsto.

À Tânia, de quem todo o carinho e amor recebidos me fizeram, nestes anos, alcançar metas de outra forma inatingíveis.

Aos meus pais e irmão, que me apoiaram e ajudaram em todos os obstáculos que apareceram e sem os quais era impensável vencer.

À Conceição, por todo o apoio que me deu.

Aos meus verdadeiros amigos.

À Mel, pela companhia que me fez ao longo das infinitas horas de trabalho.

A todos o meu sincero muito Obrigado!

Fernando

Resumo

A tecnologia WIMAX (*Worldwide Interoperability for Microwave Access*), definida na norma IEEE 802.16, tem recebido grande destaque no que diz respeito a tecnologias de acesso sem fios em banda larga (BWA – *Broadband Wireless Access*). O WIMAX oferece acessos sem fios com características específicas, tal como, o suporte de qualidade de serviço (QoS) na camada MAC. Estando esta tecnologia ainda em evolução, a implementação de soluções é, normalmente, precedida por fases de estudos e de simulações de forma a encontrar soluções que viabilizem o seu sucesso. O simulador ns-2 (*Network Simulator*) é uma das ferramentas mais utilizadas para a simulação de redes computacionais. Esta ferramenta permite a simulação de redes com e sem fios, bem como de vários protocolos utilizados na internet. No entanto, o ns-2 ainda não permite simular a tecnologia WIMAX, levando os investigadores a proporem módulos que, integrados na ferramenta ns-2, possibilitem a simulação desta tecnologia.

Este trabalho apresenta um estudo do WIMAX efectuando-se uma comparação de três módulos implementados para a simulação deste protocolo no ns-2. A norma IEEE 802.16 não especifica como deve ser controlado o escalonamento de QoS, pelo que o objectivo deste trabalho centrou-se na comparação dos módulos disponíveis, avaliando-se quais os que apresentam uma abordagem mais rigorosa ao nível do escalonamento de QoS. O primeiro módulo simulado foi desenvolvido pela *Networks & Distributed Systems Laboratory* (NDSL) tendo a capacidade de efectuar o controlo de QoS na camada MAC enquanto que, o segundo módulo, implementado pelo *National Institute of Standards and Technology*, NIST, não tem essa competência. Recentemente foi disponibilizado mais um módulo capaz de garantir a qualidade de serviço, implementado pela Universidade Estadual de Campinas (UNICAMP). Os resultados obtidos nas simulações efectuadas neste trabalho tentam comparar os módulos, verificando se estes são capazes de assegurar as especificações da norma 802.16, para todas as classes de serviços e em diferentes cenários de simulação.

Palavras-Chave: WIMAX, QoS, Escalonador, Camada MAC, NIST, NDSL, UNICAMP, NS-2.

Abstract

The WIMAX (Worldwide Interoperability for Microwave Access) technology, defined on IEEE 802.16 standard, has received great attention in the area of Broadband Wireless Access (BWA) technology. WIMAX provides wireless accesses with specific characteristics, such as, the support for Quality of Service (QoS) in the MAC Layer. Since this technology is still evolving, implementations are usually preceded by the study and the simulation steps, towards the definition of a successful solution. The Network Simulator (ns-2) is a popular tool for simulation of computer networks. This tool provides substantial support for simulation of Internet protocols over wired and wireless networks. Nevertheless, the ns-2 still does not provide facilities to simulate the WiMAX technology, leading the researchers to propose WIMAX modules, which are integrated with ns-2 in order to enable the simulation of this technology.

This thesis presents a study of WIMAX, which provides a comparison of three WIMAX modules developed for simulating this protocol with ns-2. The IEEE 802.16 standard does not specify how should be controlled the scheduling mechanisms for QoS, therefore the main purpose of this study was to compare the available modules and to assess which one implements the most rigorous approach of QoS scheduling. The first module was developed by Networks & Distributed Systems Laboratory (NDSL), and has the ability to manage QoS in the MAC layer, while a second module developed by the National Institute of Standards and Technology (NIST) has no such competence. Recently, a new module was released by the State University of Campinas (UNICAMP) that also implements the quality of service mechanisms. The results of the simulations described in this thesis aim to compare the performance of modules and to check if they are able to ensure the specifications of the 802.16 standard, for all classes of services in different simulation scenarios.

Keywords: WIMAX, QoS, Scheduler, MAC layer, NIST, NDSL, UNICAMP, NS-2.

Índice

AGRADECIMENTOS	I
RESUMO	III
ABSTRACT	V
ÍNDICE	VII
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABELAS	XIII
ACRÓNIMOS.....	1
1. INTRODUÇÃO	7
1.1. CONTEXTUALIZAÇÃO	7
1.2. OBJECTIVOS.....	8
1.3. ORGANIZAÇÃO DO RELATÓRIO	8
2. WIMAX.....	9
2.1. NORMA IEEE 802.16.....	11
2.2. WIMAX FÓRUM.....	12
2.3. ARQUITECTURA	13
2.4. ACESSO FIXO WIRELESS.....	14
2.5. ACESSO MÓVEL.....	17
2.6. CONTROLO DO ACESSO AO MEIO – MAC (<i>MEDIUM ACCESS CONTROL</i>)	18
2.7. SUBCAMADA DE SEGURANÇA	31
2.8. CAMADA FÍSICA.....	32
2.9. RESUMO.....	35
3. INVESTIGAÇÃO DE QOS EM WIMAX.....	37
3.1. “ <i>ENSURING THE QoS REQUIREMENTS IN 802.16 SCHEDULING</i> ” DE ALEXANDER SAYENKO ET AL	38
3.2. “ <i>PERFORMANCE ANALYSIS OF THE IEEE 802.16 WIRELESS METROPOLITAN AREA NETWORK</i> ” DE DONG-HOON CHO ET AL.....	43
3.3. “ <i>AN INTEGRATED QoS CONTROL ARCHITECTURE FOR IEEE 802.16 BROADBAND WIRELESS ACCESS SYSTEMS</i> ” DE JIANFENG CHEN ET AL	46
3.4. “ <i>DESIGN AND IMPLEMENTATION OF A QoS-INCLUDED WIMAX MODULE FOR NS-2 SIMULATOR</i> ” DE AYMEN BELGHITH E LOUTFI NUAYMI	50
3.5. RESUMO.....	60
4. FERRAMENTA DE SIMULAÇÃO – NS	63
4.1. PORQUÊ O NS?	63

4.2.	INTRODUÇÃO.....	64
4.3.	FUNCIONAMENTO GERAL	65
4.4.	PROCESSO DE SIMULAÇÃO.....	67
4.5.	<i>ROUTING</i>	73
4.6.	ANÁLISE DOS RESULTADOS	73
4.7.	AWK	75
4.8.	CONFIGURAÇÃO DE CENÁRIOS DE SIMULAÇÃO.....	77
4.9.	RESUMO	82
5.	DESCRIÇÃO DOS MÓDULOS DE WIMAX PARA O NS-2	83
5.1.	MÓDULO NDSL	84
5.2.	MÓDULO NIST.....	94
5.3.	MÓDULO UNICAMP	105
5.4.	FUNCIONALIDADES DOS MÓDULOS	113
5.5.	RESUMO	115
6.	SIMULAÇÃO & ANÁLISE DE RESULTADOS.....	117
6.1.	MEDIÇÕES UTILIZADAS	120
6.2.	ANÁLISE DOS VALORES PUBLICADOS	121
6.3.	1º CENÁRIO DE SIMULAÇÃO – ESTUDO DAS DIFERENTES CLASSES DE SERVIÇO	124
6.4.	2º CENÁRIO DE SIMULAÇÃO – ESTUDO DO COMPORTAMENTO DO ESCALONADOR PARA CLASSES DE ALTA PRIORIDADE.....	138
6.5.	3º CENÁRIO DE SIMULAÇÃO – ESTUDO DO COMPORTAMENTO DO ESCALONADOR PARA A CLASSE DE MENOR PRIORIDADE (BE)	143
6.6.	RESUMO	152
7.	CONCLUSÕES	155
	REFERÊNCIAS DOCUMENTAIS.....	159

Índice de Figuras

Figura 1	- Vista de diferentes dimensões de redes [2]	10
Figura 2	- Modelo de referência da rede WIMAX [13]	13
Figura 3	- Arquitectura PTP em FWA	15
Figura 4	- Arquitectura PMP em FWA	16
Figura 5	- Arquitectura <i>mesh</i> em FWA	16
Figura 6	- Norma 802.16 e a sua evolução [5]	18
Figura 7	- Camadas da norma IEEE 802.16	18
Figura 8	- Classificação e mapeamento CID [6]	20
Figura 9	- Formato MAC SDU [6]	20
Figura 10	- Estrutura da trama TDD [6]	22
Figura 11	- Estrutura da trama OFDM [6]	23
Figura 12	- DL-MAP feitas para <i>bursts da subtrama de downlink</i> de vários tipos [6]	24
Figura 13	- DL-MAP e UL-MAP em TDD [6]	24
Figura 14	- Procedimento para entrada na rede de uma SS	26
Figura 15	- Formato de um MAC PDU [6]	27
Figura 16	- Arquitectura possível de QoS da norma 802.16 [9]	30
Figura 17	- OFDM e OFDMA [8]	34
Figura 18	- Estrutura da rede (cenário 1) [19]	40
Figura 19	- Resultados obtidos – Throughput – taxa de transferência (cenário 1) [19]	40
Figura 20	- Estrutura da rede (cenário 2) [19]	41
Figura 21	- Resultados obtidos – <i>Throughput</i> – taxa de transferência (cenário 2) [19]	41
Figura 22	- Estrutura da rede (cenário 3) [19]	42
Figura 23	- Resultados obtidos – <i>Throughput</i> – taxa de transferência (cenário 3) [19]	43
Figura 24	- Arquitectura de QoS proposta [20]	44
Figura 25	- Utilização do canal quando não existe limite de largura de banda para o tráfego de maior prioridade [9]	45
Figura 26	- Utilização do canal quando uma cota fixa da largura de banda é permitida ao fluxo UGS [9]	46
Figura 27	- Arquitectura <i>multi-layer</i> de controlo de QoS integrados [21]	48
Figura 28	- Tempo Setup vs taxa de chegada [21]	49
Figura 29	- Fluxo do serviço de entrada vs curva de serviço [21]	49
Figura 30	- Passos para a criação de um novo fluxo de serviço [22]	51
Figura 31	- Oportunidade de pedidos <i>unicast</i> [22]	52
Figura 32	- Oportunidade de pedidos por contenção [22]	53

Figura 33	- Escalonamento UGS [22].....	54
Figura 34	- Escalonamento rtPS [22].....	55
Figura 35	- Escalonamento BE [22].....	56
Figura 36	- Taxa de transferência efectiva rtPS vs Carga do tráfego rtPS oferecida [22]	57
Figura 37	- Tempo de atraso (<i>mean sojourn time</i>) rtPS vs carga do tráfego rtPS oferecida.....	58
Figura 38	- Taxa de transferência efectiva UGS vs Carga do tráfego UGS oferecida [22]	58
Figura 39	- Tempo de atraso (<i>mean sojourn time</i>) UGS vs carga do tráfego UGS oferecida	59
Figura 40	- Taxa de transferência efectiva UGS vs Carga do tráfego UGS oferecida [22]	60
Figura 41	- C++ e OTcl [27].....	65
Figura 42	- Vista simplificada do ns-2 do ponto de vista do utilizador	66
Figura 43	- Vista da arquitectura geral do NS	67
Figura 44	- Escalonador de eventos discretos [27]	68
Figura 45	- Estrutura interna de um nó <i>unicast</i> [28]	69
Figura 46	- Estrutura interna de um nó <i>multicast</i> [28]	69
Figura 47	- Composição de uma ligação unidireccional [28]	71
Figura 48	- Formato de um pacote no ns [27].....	72
Figura 49	Ex de um <i>trace file</i> [29].....	74
Figura 50	- NAM – <i>network animator</i>	75
Figura 51	- Simples topologia de rede e o cenário de simulação [27]	78
Figura 52	- Relação entre o módulo WIMAX e os módulos do ns [33]	87
Figura 53	- Arquitectura da camada MAC simulada da norma IEEE 802.16 [33]	88
Figura 54	- Esquema do método Weighted Round-Robin [34]	91
Figura 55	- Diagrama da classe <i>Classifier</i> [36]	96
Figura 56	- Diagrama da classe MAC 802.16 [36]	96
Figura 57	Entrada na rede [36]	99
Figura 58	- Processo de <i>scanning</i> [36].....	100
Figura 59	- Processamento dos pacotes de saída [36].....	101
Figura 60	- Topologia do cenário criado.....	105
Figura 61	- Estrutura do módulo para a simulação na norma IEEE 802.16 [38]	106
Figura 62	- Arquitectura do mecanismo de gestão proposto [39].....	109
Figura 63	- Cenário 1 BS e 5 SS com diferentes tráfegos a serem gerados	119
Figura 64	- Atraso médio em função do nº de SSs (NDSL)	121
Figura 65	- Atraso médio em função do nº de SSs b)	122
Figura 66	- Atraso das ligações rtPS e UGS (UNICAMP)	123
Figura 67	- Atraso das ligações rtPS e UGS b).....	123
Figura 68	- Estrutura de rede – 1º cenário (análise I)	127
Figura 69	- Taxa de transferência efectiva de 5SSs (NDSL)	128
Figura 70	- Taxa de transferência efectiva de 5SSs (UNICAMP)	130
Figura 71	- Taxa de transferência efectiva de 5SSs b) (UNICAMP).....	130

Figura 72	- Taxa de transferência efectiva de 5SSs c) (UNICAMP).....	131
Figura 73	- Atrasos referentes ao 1º cenário, análise I	132
Figura 74	- Estrutura da rede – 1º cenário (análise II)	134
Figura 75	- Taxa de transferência efectiva para 5 SSs (NDSL).....	135
Figura 76	- Taxa de transferência efectiva para 5 SSs (UNICAMP).....	136
Figura 77	- Taxa de transferência efectiva SS5 (BE) – UNICAMP	137
Figura 78	- Atrasos referentes ao 1º cenário, análise II	137
Figura 79	- Estrutura da rede – 2º cenário (análise I)	139
Figura 80	- Taxa de transferência de uma ligação UGS	139
Figura 81	- Atrasos médios da classe UGS.....	140
Figura 82	- Estrutura de rede – 2º cenário (análise II)	141
Figura 83	- Taxa de transferência efectiva – análise II.....	141
Figura 84	- Atrasos médios da classe rtPS.....	142
Figura 85	- Estrutura de rede – 3º cenário (análise I)	143
Figura 86	- Taxa de transferência efectiva para uma SS (BE).....	144
Figura 87	- Estrutura da rede – 3º cenário (análise II)	145
Figura 88	- Taxa de transferência para 15 SSs b) (NDSL)	145
Figura 89	- Taxa de transferência efectiva para 15 SSs b) (UNICAMP)	146
Figura 90	- Taxa de transferência efectiva para 15 SSs – módulo NIST.....	147
Figura 91	- Taxa de transferência para 15 SSs – 3º cenário (análise III) (NDSL).....	149
Figura 92	- Taxa de transferência para 15 SSs (UNICAMP)	150
Figura 93	- Taxa de transferência para 15 SSs (NIST).....	151

Índice de Tabelas

Tabela 1	- Normas 802.16 [9]	11
Tabela 2	- Grupos de trabalho do WIMAX fórum [12]	12
Tabela 3	- Classes de QoS.....	30
Tabela 4	- Limites da taxa de taxas de transferências impostas [19]	42
Tabela 5	- Modulações utilizadas [19]	42
Tabela 6	- Explicação do <i>trace file</i> [29].....	74
Tabela 7	- Funcionalidades / limitações do módulo da NIST	95
Tabela 8	- Mensagens de controlo definidas.....	98
Tabela 9	- Comparação dos módulos NDSL, UNICAMP e NIST.....	114
Tabela 10	- Requisitos das classes de QoS	125
Tabela 11	- Aplicações das SSs	127
Tabela 12	- Aplicações activas.....	148

Acrónimos

AAS	–	Adaptative Antenna System
ABP	–	Adaptive Burst Profiles
AES	–	Advanced Encryption Standard
AODV	–	Ad hoc on Demand Distance Vector routing
API	–	Application Programming Interface
ARQ	–	Automatic Repeat Request
ASN	–	Access Service Network
ATM	–	Asynchronous Transfer Mode
BE	–	Best Effort
BS	–	Base Station
BR	–	Bandwidth Request
BWA	–	Broadband Wireless Access
BFWA	–	Broadband Fixed Wireless Access
CBQ	–	Class Based Queueing
CBR	–	Constant Bit Rate
CDMA	–	Code Division Multiple Access
CID	–	Connection Identifiers
CN	–	Core Network
CPS	–	Common Part Sublayer
CRC	–	Cyclic Redundancy Check
CS	–	Convergence Sublayer
CSMA	–	Carrier Sense Multiple Access
CSN	–	Connectivity Service Network

DAMA	– Demand Assigned Multiple Access
DCD	– Downlink Channel Descriptor
DES3	– Triple Data Encryption Standard
DF	– Deadline First
DFPQ	– Deficit Fair Priority Queue
DHCP	– Dynamic Host Configuration Protocol
DIUC	– Downlink Interval Usage Code
DL	– Downlink
DOCSIS	– Data Over Cable Service Interface Specification
DRR	– Deficit Round Robin
DSA	– Dynamic Service Addition
DSC	– Dynamic Service Change
DSCP	– Differentiated Services Code Point
DSDV	– Destination Sequence Distance Vector
DSL	– Digital Subscriber Line
DSR	– Dynamic Source Routing
ertPS	– extended real time Polling Service
ETSI	– European Telecommunications Standards Institute
FBSS	– Fast Base Station Switching
FCH	– Frame Control Header
FDD	– Frequency Division Duplex
FEC	– Forward Error Correction
FIFO	– First in First Out
FQ	– Fair Queuing
FSM	– Finite State Machine
FTP	– File Transfer Protocol
FWA	– Fixed Wireless Access

GSM	– Global System for Mobile Communications
HHO	– Hard Handover
HTTP	– Hypertext Transfer Protocol
IE	– Information Elements
IEEE	– Institute of Electrical and Electronics Engineers
IETF	– Internet Engineering Task Force
IFQ	– Interface Queue
IP	– Internet Protocol
ISP	– Internet Service Provider
LAN	– Local Area Network
LL	– Link Layer
LOS	– Line of Sight
MAC	– Medium Access Control
MAN	– Metropolitan Area Network
MDHO	– Macro Diversity Handover
MPEG	– Moving Picture Experts Group
MS	– Mobile Station
MSDU	– MAC Service Data Unit
mSIR	– maximum Signal-to-Interference Ratio
MSS	– Mobile SS
NAM	– Network Animator
NAP	– Network Access Provider
NDSL	– Networks & Distributed Systems Laboratory
NIST	– National Institute of Standards and Technology
NLOS	– Non Line of Sight
nrtPS	– Non Real Time Polling Services
NS	– Network Simulator

NSP	– Network Service Provider
OFDM	– Orthogonal Frequency Division Multiplexing
OFDMA	– Orthogonal Frequency Division Multiple Access
OPNET	– Optimized Network Engineering Tools
OSI	– Open Systems Interconnection
OTcl Object	– Oriented Tcl
PDU	– Packet Data Unit
PHB	– Per-hop Behaviors
PHS	– Package Header Suppression
PHSF	– PHS Field
PHSI	– PHS Index
PHY	– Physical
PKM	– Key Management Protocol
PMP	– Point to Multipoint
PQ	– Priority Queue
PS	– Physical Slots
PTP	– Point to Point
QAM	– Quadrature Amplitude Modulation
QoS	– Quality of Service
RED	– Random Early Detection
REG-RSP	– Registration Response
RLC	– Radio Link Control
RNG-RSP	– Ranging Response
RR	– Round Robin
RRM	– Radio Resource Management
RTG	– Receive/transmit Transition Gap
RTP	– Run Time Protocol

rtPS	– Real Time Polling Service
SA	– Security Association
SAP	– Service Access Point
SC	– Single Carrier
SDU	– Service Data Unit
SFID	– Service Flow Identifier
SFQ	– Stochastic Fair Queueing
SNR	– Signal-to-Noise Ratio
SS	– Subscriber Station
TCID	– Transport CID
Tcl	– Tool Comment Language
TCP	– Transmission Control Protocol
TDD	– Time Division Duplex
TDMA	– Time Division Multiple Access
TFTP	– Trivial File Transfer Protocol
TGA	– Traffic Generating Agent
TKIP	– Temporal Key Integrity Protocol
TLV	– Type/Length/Value
TORA	– Temporally-Ordered Routing Algorithm
TRS	– Temporary Removal Scheduler
TTG	– Transmit/receive Transition Gap
TTL	– Time To Live
UCD	– Uplink Channel Descriptor
UDP	– User Datagram Protocol
UGS	– Unsolicited Grant Service
UIUC	– Uplink Interval Usage Code
UL	– Uplink

UML	– Unified Modeling Language
UMTS	– Universal Mobile Telecommunications
UNICAMP	– Universidade Estadual de Campinas
VoIP	– Voice over IP
W ² FQ	– Worst-Case Fair Queuing
WAN	– World Area Network
WFQ	– Weighted Fair Queuing
WG	– Working Groups
Wi-Fi	– Wireless Fidelity
WiMAX	– Worldwide Interoperability for Microwave Access
WirelessMAN	– Wireless Metropolitan Area Network
WirelessHUMAN	– Wireless High-Speed Unlicensed Metropolitan Area Network
WLAN	– Wireless Local Area Network
WRR	– Weighted round robin

1. INTRODUÇÃO

1.1. CONTEXTUALIZAÇÃO

Este projecto surgiu da necessidade de realização de um estudo sobre a tecnologia emergente WIMAX (*Worldwide Interoperability for Microwave Access*), padronizada pela norma IEEE 802.16.

O acesso sem fios em banda larga (*Broadband Wireless Access* - BWA) tem recebido grande destaque tanto ao nível comercial como ao nível académico, por ser uma tecnologia capaz de promover altas taxas de transferências de dados, alto nível de escalabilidade e baixo custo de instalação e manutenção dos acessos à internet.

A norma 802.16 foi desenvolvida com o intuito de regular a tecnologia BWA definindo tanto a interface aérea de rádio como o protocolo de acesso ao meio. Esta norma deixa em aberto vários aspectos de implementação para que os fabricantes possam diferenciar os seus produtos. Um dos pontos mais interessantes, deixado em aberto, é a forma como serão elaborados os mecanismos para garantir a Qualidade de Serviço (QoS), realizados na direcção das estações clientes (*Subscriber Stations* – SSs) para a estação base (*Base Station* – BS), isto é no sentido do *uplink*, na camada de acesso ao meio (*Medium Access Control* - MAC). No caso do acesso fixo wireless é importante existirem mecanismos de escalonamento para o tráfego de *uplink*, enquanto no tráfego de *downlink*, isto é no sentido

inverso, estes mecanismos não são tão relevantes. No tráfego de *downlink*, a BS é a única estação que transmite dados, logo não haverá a necessidade de controlar a concorrência na requisição da largura de banda. Os pacotes são enviados por *broadcast* a todas as SSs, pertencendo a cada SS o papel de receber quais são os pacotes destinados a si. No caso do tráfego de *uplink* há concorrência entre as SSs para o envio dos pacotes. Neste caso será aconselhável que a BS possua mecanismos de gestão para que os parâmetros de QoS sejam satisfeitos. Assim, para os utilizadores, a qualidade do serviço será garantida em qualquer tipo de aplicação.

1.2. OBJECTIVOS

O objectivo principal deste projecto é efectuar um estudo intensivo da tecnologia WIMAX, aprofundando a investigação no âmbito dos mecanismos de implementação das classes de QoS.

Com o intuito de estudar os diversos mecanismos de gestão de QoS do WIMAX em cenários de simulação, seleccionou-se uma ferramenta de simulação de redes, disponível em software aberto, o *Network Simulator* (ns-2).

Assim, com este trabalho pretende-se identificar e caracterizar, tanto do ponto de vista funcional como de eficiência, um conjunto de implementações de módulos WIMAX para ns-2, disponíveis gratuitamente, que permitam simular esta tecnologia do modo mais rigoroso possível.

1.3. ORGANIZAÇÃO DO RELATÓRIO

O capítulo 1 é composto pela introdução do trabalho. No capítulo 2 é apresentada a tecnologia WIMAX, com particular ênfase na camada MAC onde são introduzidos os mecanismos de gestão de QoS. No capítulo 3, é feita uma apresentação de trabalhos científicos propostos para gestão das classes de QoS. No capítulo 4, é apresentada a ferramenta de simulação, o ns-2. O 5º capítulo é composto por 3 subcapítulos, onde são explicados os três módulos da tecnologia WIMAX feitos para o ns-2 e ainda um subcapítulo, onde é efectuada a comparação das vantagens e inconvenientes de cada módulo. O capítulo 6 descreve a simulação e análise dos resultados obtidos. No capítulo 7 são apresentadas as principais conclusões e perspectivados futuros desenvolvimentos.

2. WIMAX

Nas tecnologias *wireless* é comum existirem limitações tanto ao nível da área de cobertura como também ao nível da largura de banda. Desta forma, o WIMAX (*Worldwide Interoperability for Microwave Access*) é a tecnologia que tem como propósito resolver as restrições de banda existente, fornecendo um acesso *wireless* em banda larga capaz de abranger uma grande área de cobertura. Juntamente com estas funcionalidades o WIMAX está associado a custos relativamente baixos de instalação e manutenção. A sua utilização pode ser efectuada na rede de acesso permitindo aos operadores de telecomunicações oferecer serviços de conectividade tanto em locais da área metropolitana como em zonas rurais onde a cablagem de infra-estrutura não existe. Para países ainda em desenvolvimento esta tecnologia apresenta-se também como uma alternativa muito atractiva.

O WIMAX é uma tecnologia sem fios de banda larga que se direcciona para oferecer um serviço na escala da *Metropolitan Area Network* (MAN) podendo atingir um alcance até 50km e uma taxa de transmissão de 70 Mbps [1]. As redes MAN são redes de dados desenvolvidas através de infra-estruturas de alta velocidade, capazes de servir uma área de dimensão equivalente a uma cidade. A Figura 1 ilustra diferentes dimensões de redes existentes.

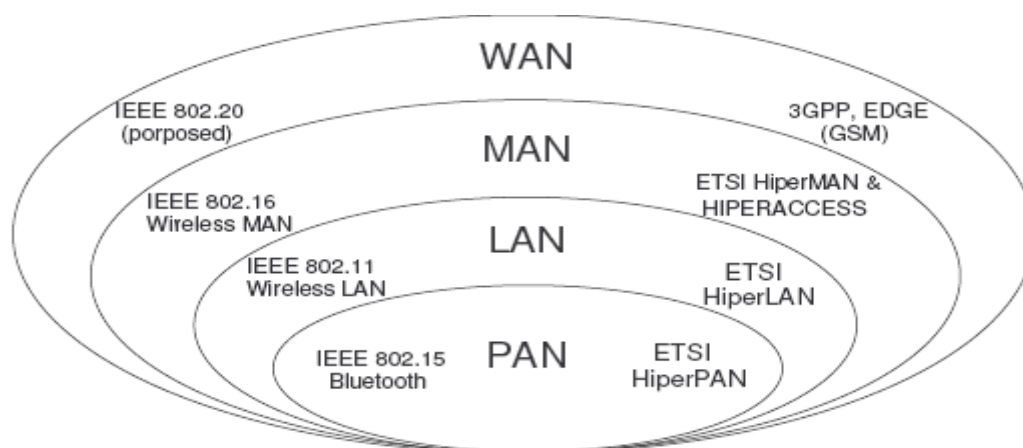


Figura 1 - Vista de diferentes dimensões de redes [2]

O WIMAX enquadra-se entre a WAN (*World Area Network*), que apresenta uma cobertura global, e a LAN (*Local Area Network*), de menor dimensão, apresentando uma cobertura capaz de servir áreas de pequena dimensão como prédios ou conjunto de prédios.

Associada à LAN está a WLAN (*Wireless LAN*) que é da mesma escala que a LAN mas usa transmissões wireless via rádio em substituição do cabo. A norma 802.11, comumente referenciada como WI-FI, está associada à escala WLAN. Efectuando um paralelismo entre WI-FI e WIMAX, é possível afirmar que o WIMAX, padronizado pela norma 802.16, está para a MAN assim como o WI-FI está para a LAN, apresentando velocidades de transmissão muito mais atraentes [3][4].

O WIMAX baseia-se na norma IEEE 802.16, uma norma em evolução que se iniciou no *National Institute of Standards and Technologies* (NIST) em 1998 antes de ser transferida para o *Institute of Electrical and Electronics Engineers* (IEEE) para formar o grupo de trabalho 802.16.

Em Junho de 2004, este grupo de trabalho conseguiu que fosse aprovada uma versão da norma 802.16 para o acesso fixo *wireless*, conhecido por IEEE 802.16-2004. O primeiro laboratório certificado, estabelecido na Cetecom Labs em Málaga, Espanha, está em pleno funcionamento e mais de 150 experiências WIMAX estão em desenvolvimento na Europa, Ásia, África e América. Todos estes testes têm provado que este acesso fixo *wireless* possui uma óptima relação preço-eficácia fazendo com que este seja uma forte alternativa aos serviços por cabo ou DSL [5].

Em Dezembro de 2005, uma extensão dirigida à mobilidade e portabilidade também conseguiu a aprovação pelo IEEE, ficando conhecida com IEEE 802.16e-2005 [1].

2.1. NORMA IEEE 802.16

A versão IEEE 802.16-2004 [6] (também conhecida por 802.16d) inclui aspectos definidos em várias versões anteriores (802.16-2001, 802.16c em 2002 e 802.16a em 2003). As alterações introduzidas nestas normas centraram-se nas aplicações fixas ou nómadas, a operar na frequência de 2-11 GHz disponibilizando uma largura de banda de 70 Mbps ou 2-10 Mbps/utilizador numa área até 10 km² (teoricamente pode cobrir uma distância até 50km). A norma define quatro classes de serviço capazes de assegurar a qualidade de serviço (QoS) necessária para o bom funcionamento de qualquer tipo de tráfego.

A versão IEEE 802.16e [7] acrescenta um suporte *roaming* a velocidades de veículos, promovendo largura de banda de 1-3 Mbps/utilizador e um alcance de 5km, introduzindo o conceito de mobilidade, sendo este o aspecto principal para o aparecimento desta norma. Uma das inovações trazidas por esta nova reforma foi o aparecimento de uma quinta classe de QoS [8]. A Tabela 1 traduz a evolução da norma IEEE 802.16.

Tabela 1 - Normas 802.16 [9]

	A família IEEE 802.16
802.16	Padrão para BWA que opera em frequências entre 10 e 66 GHz. Necessita de linha de visão.
802.16a	Actualiza o padrão 802.16 para operar em frequências de 2 a 11 GHz; Alcance de 50 km; Não necessita de linha de visão.
802.16b	Aplicações que permite o uso de frequências de 5 a 6 GHz não licenciadas.
802.16c	Interoperabilidade das frequências até 66 GHz não licenciadas.
802.16d	Melhoramento do 802.16, 802.16a, 802.16c tornando-os desactualizados e obsoletos.
802.16e	Introduz suporte a uma mobilidade até 60 km.
802.16f	Evolução do 802.16 introduzindo o conceito de redes em malha (<i>mesh networks</i>).
802.16g	Outra evolução para suporte de mobilidade.

2.2. WIMAX FÓRUM

O WIMAX fórum é uma organização sem fins lucrativos, fundada em 2001, que acompanha o desenvolvimento desta tecnologia nas suas diversas vertentes. Os membros do WIMAX fórum são uma mistura de pequenas e grandes empresas tendo todas o mesmo propósito, fazer parte das tentativas de standardização e de certificação [10].

O trabalho realizado pelo WIMAX fórum consiste em assegurar a interoperabilidade e conformidade entre os equipamentos de acesso wireless *broadband* baseados nas normas IEEE 802.16 e ETSI (*European Telecommunications Standards Institute*) [11]. Estas empresas juntam os seus esforços para combinarem as duas normas, criando um processo de certificação para produtos que operam com a tecnologia WIMAX.

O WIMAX fórum é organizado em grupos de trabalho (WG – Working Groups). A Tabela 2 relata os propósitos de cada um destes grupos de trabalho.

Tabela 2 - Grupos de trabalho do WIMAX fórum [12]

Grupos de trabalho	Propósito
Aplicação	Define aplicações sobre a tecnologia WIMAX necessárias para satisfazer conjuntos de ofertas competitivas e que estão excepcionalmente reforçadas por esta tecnologia.
Certificação	Lida com os aspectos operacionais do programa de certificação do WIMAX Fórum; interfaces com laboratórios de certificação; selecção de novos laboratórios de certificação.
Marketing	Promove o WIMAX Fórum.
Rede	Cria especificações de rede de alto nível para sistemas WIMAX fixos, móveis, portáteis e nómadas. Define a arquitectura da rede WIMAX.
Regulação	Influencia as agências reguladoras globais a promover alocações espectrais WIMAX-friendly.
Fornecedor de serviço	Oferece aos prestadores de serviço uma plataforma para influenciar o produto BWA e requisitos de espectro, garantindo que as suas necessidades individuais de mercado sejam preenchidas.
Técnico	Desenvolve testes de conformidade de especificações e de certificação de serviços para atingir uma interoperabilidade à escala mundial

2.3. ARQUITECTURA

A arquitectura de uma rede WIMAX possui dois elementos principais: Estação Base (BS – *Base Station*) e Estação do Assinante (SS – *Subscriber Station*). A BS realiza a comunicação entre os acessos através da rede wireless e a rede do operador suportando interfaces IP, ATM, Ethernet e E1/T1. A SS fornece ao utilizador acesso à rede *core* do operador através do estabelecimento de ligações com a BS em várias topologias.

Porque a norma IEEE 802.16 se limita às camadas físicas e de acesso ao meio, o WIMAX fórum desenvolveu uma arquitectura de rede *end-to-end* especificando os sistemas acesso/*core* e as suas funcionalidades. A norma define procedimentos e protocolos para a rede suportar, por exemplo, mobilidade, segurança, *internetworking* e autenticação para uma SS do WIMAX. Uma descrição da arquitectura de rede é apresentada no modelo de referência na Figura 2 que contém entidades tais como *Subscriber Station* (SS), *Access Service Network* (ASN), onde se situam as BSs e *Connectivity Service Network* (CSN) [13].

Este modelo de referência também contém as interfaces entre diferentes entidades. Estas interfaces definem procedimentos e protocolos fornecendo também ligações lógicas em detrimento de ligações físicas entre as entidades.

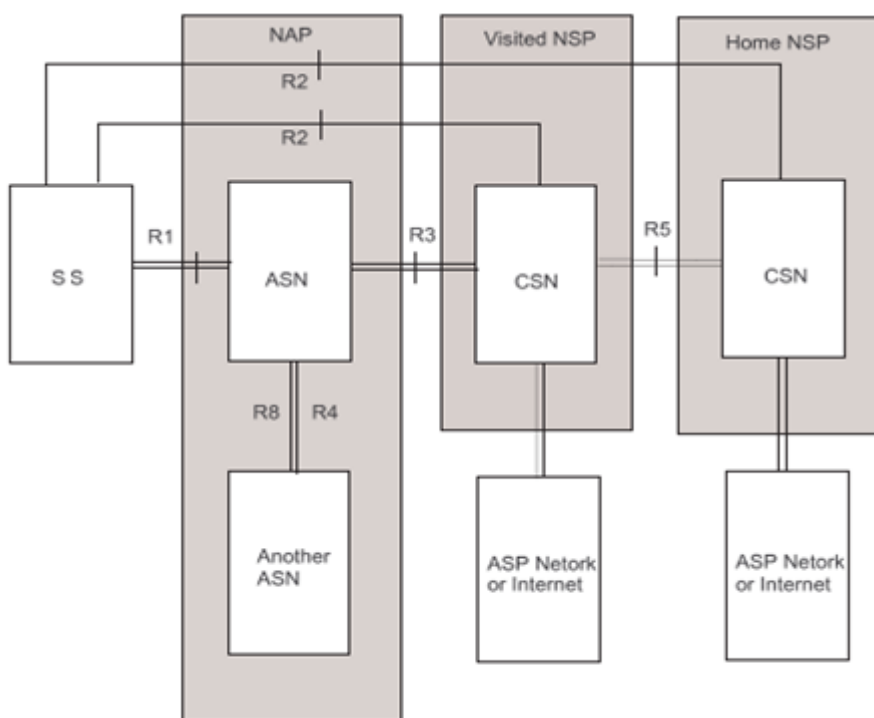


Figura 2 - Modelo de referência da rede WIMAX [13]

2.3.1. REDE DE ACESSO AO SERVIÇO

A rede de acesso ao serviço (ASN – *Access Service Network*) consiste num ou vários *gateways* ASN e BSs, fornecendo uma cobertura via rádio a uma determinada área geográfica. Uma ASN controla funcionalidades de acesso da camada MAC tais como paginação, localização, controlo de recursos rádio (RRM – *Radio Resource Management*) e mobilidade entre BSs. A ASN controla apenas as ligações rádio do WIMAX, deixando o controlo de alto nível para o CSN. A ASN pode também ser usada como uma *proxy*.

A ASN é desdobrada por uma entidade de negócio chamada *Network Access Provider* (NAP) que fornece conectividade entre uma M/SS e uma rede rádio WIMAX ligando os utilizadores ao *Network Service Provider* (NSP) que controla a rede de ligação (CSN). A *gateway* ASN interliga a ASN e a CSN.

2.3.2. REDE DE LIGAÇÃO AO SERVIÇO

Uma rede de ligação ao serviço (CSN – *Connectivity Service Network*) é um conjunto de funções de rede que fornece ligações IP a uma SS. A CSN contém *gateways* para o acesso à internet, routers, servidores, *proxies*, bases de dados de utilizadores e dispositivos de internet. Também possui uma política de controlo e de admissão, mobilidade entre ASNs e serviços específicos de WIMAX, como por exemplo, serviços de verificação do cumprimento da lei. A CSN é composta por uma entidade negociadora chamada NSP. As SSs entram em acordo contratual no que diz respeito a QoS, largura de banda, etc., com a NSP fazendo o acesso destes serviços pela rede de acesso (ASN) onde está inserida.

2.4. ACESSO FIXO WIRELESS

O acesso fixo wireless (FWA – *Fixed Wireless Access*) tem como principal propósito a substituição das ligações “*last-mile*” existentes entre os utilizadores e o núcleo da rede servindo-se de um ambiente wireless. A designação “fixo” deve-se ao facto dos emissores e receptores serem ambos estacionários contrastando com outras redes de comunicação wireless onde estes são móveis. O desenvolvimento de redes FWA permite aos utilizadores um acesso imediato à rede sem necessidade de efectuar a passagem de cabos subterrâneos, tornando este processo bastante menos moroso comparado com os acesos à rede efectuados por cabo [14]. O grande competidor dos sistemas FWA é a tecnologia DSL (*Digital*

Subscriber Line) [15] que usa proveito da rede pública telefónica já existente facilitando a instalação desta tecnologia.

Para o FWA, o WIMAX suporta várias topologias arquitectónicas, incluindo a arquitectura ponto-a-ponto (PTP – *Point-to-Point*), ponto-a-multiponto (PMP – *Point-to-Multipoint*) e multiponto-a-multiponto também designado por arquitectura em rede (*mesh*).

- Os sistemas PTP possuem duas estações bases com a comunicação configurada estaticamente havendo apenas uma ligação entre as duas BSs. Esta ligação dedicada é caracterizada por níveis de largura de banda superiores aos existentes noutros sistemas PTP podendo ser utilizadas antenas direccionais. Este sistema é usualmente usado para o transporte dos dados até ao núcleo da rede [14]. A Figura 3 ilustra um sistema PTP típico.

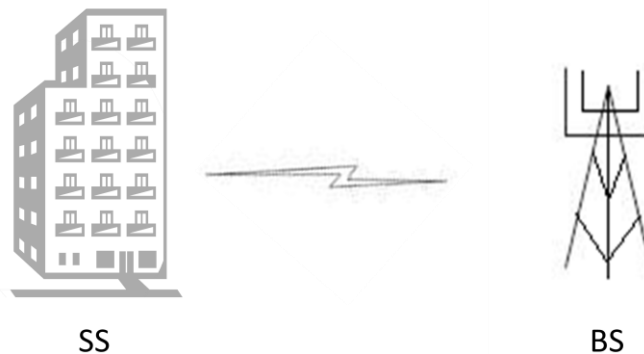


Figura 3 - Arquitectura PTP em FWA

- O sistema ponto-a-multiponto (PMP) caracteriza-se por uma estação base (BS) mais poderosa e várias estações assinantes (SSs) de menor dimensão. Os utilizadores podem aceder à rede imediatamente após instalarem o equipamento de utilizador. Neste sistema as SSs podem implantar antenas directas em direcção à BS que possui estas antenas omnidireccionais ou grupos de antenas dirigidas [14]. A Figura 4 apresenta uma configuração genérica PMP.

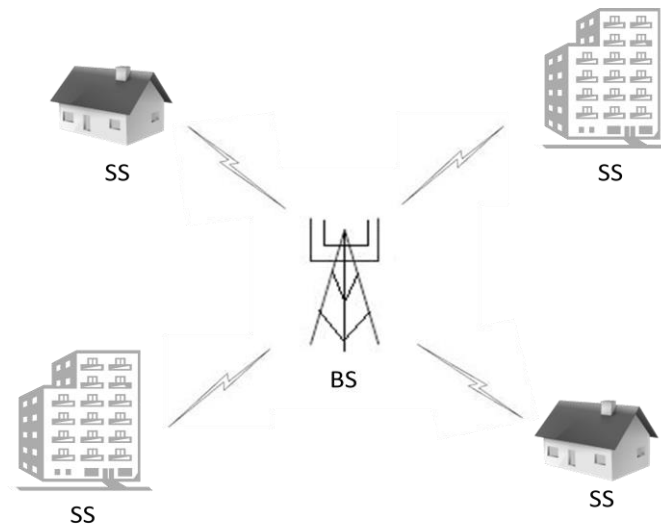


Figura 4 - Arquitectura PMP em FWA

- Em sistemas multiponto-a-multiponto, *mesh*, não existe centralização da BS. Quanto mais SSs entrarem na rede maior será a área de cobertura. Neste caso as SSs podem comunicar directamente entre si (Figura 5). A estação que não possuir linha directa de sinal entre a BS e a SS (LOS – “*line-of-sight*”) consegue receber o tráfego de outra estação através do equipamento de um outro assinante [14].

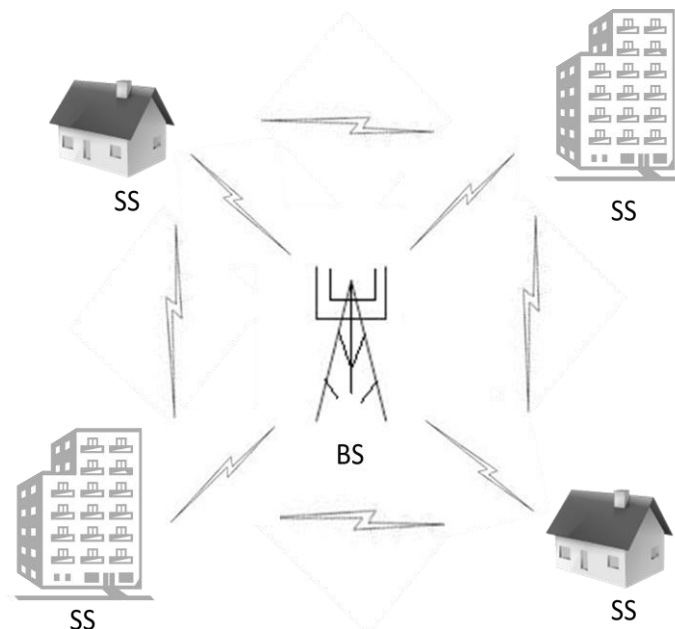


Figura 5 - Arquitectura *mesh* em FWA

2.5. ACESSO MÓVEL

O *Mobile WIMAX* suporta esquemas otimizados de *Handover* com latências abaixo dos 50 mseg para assegurar aplicações em tempo real como o VoIP, sem uma degradação do serviço [16]. O termo *handover* refere-se à transferência de uma ligação entre duas células. Este factor é determinante para os atrasos que se verificam quando se efectua, por exemplo, uma chamada num telemóvel. No caso do WIMAX o *handover* existirá quando se dá uma troca entre BSs que serve a estação móvel.

O tempo de vida da bateria e o *handover* são duas questões essenciais quando se refere a aplicações móveis. O *Mobile WIMAX* suporta um modo de adormecimento (*sleep mode*) e um modo inactivo (*idle mode*) para permitir uma operação capaz e eficiente da MS. Para além destes dois modos o *Mobile WIMAX* também possui um *handover* “sem quebra”, permitindo que a MS comute de uma BS para outra, deslocando-se à velocidade de um veículo, sem interromper a ligação [5].

Para gerir a capacidade da bateria o *mobile WIMAX* suporta dois modos – *Sleep Mode* e *Idle Mode*. No *Sleep Mode*, do ponto de vista da BS, a estação móvel não está disponível para o tráfego de DL (*DownLink*) e UL (*UpLink*). O objectivo do *Sleep Mode* é minimizar o uso da bateria da MS bem como minimizar o uso da interface aéreo que uma BS esteja a servir. Este modo também fornece a possibilidade da MS poder procurar por outras BS para recolher informações para assistir o *handover* durante o *Sleep Mode*.

O *Idle Mode* fornece um mecanismo para a MS se tornar periodicamente disponível para uma mensagem de tráfego de *broadcast* sem que a MS tenha de se registar numa BS específica à medida que a MS atravessa uma ligação aérea povoada de múltiplas BSs. Este modo beneficia a rede e as BS eliminando a interface aérea e os *handovers* na rede provocados por MSs inactivas. As MSs fornecem curtos e simples métodos para alertar as BS sobre tráfego DL que esteja ainda pendente.

Existem 3 métodos de *Handover* que a norma 802.16e suporta – *Hard Handover* (HHO), *Fast Base Station Switching* (FBSS) e *Macro Diversity Handover* (MDHO) [17].

Destes três métodos apenas o HHO é imperativo sendo os outros dois opcionais. O WIMAX Fórum desenvolveu várias técnicas para otimizar o *hard handover* dentro da estrutura da norma 802.16e. Estes desenvolvimentos têm o objectivo de reduzir os atrasos

no *handover* para menos de 50 ms [5]. A Figura 6 apresenta o conceito de mobilidade apresentado na norma IEEE 802.16e.

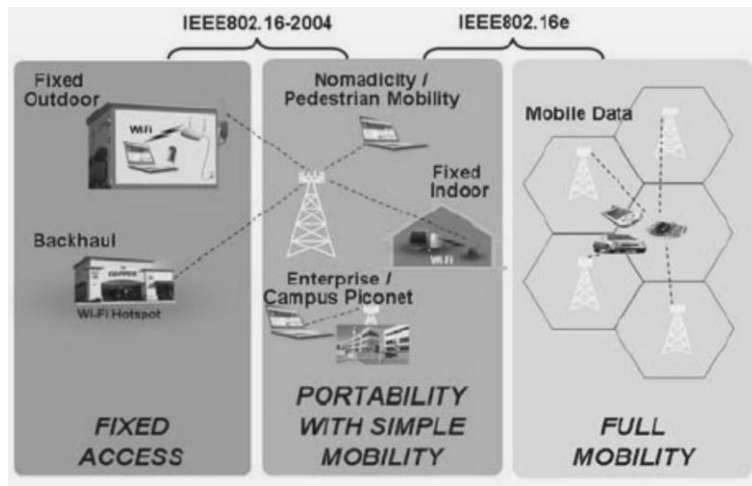


Figura 6 - Norma 802.16 e a sua evolução [5]

2.6. CONTROLO DO ACESSO AO MEIO – MAC (*MEDIUM ACCESS CONTROL*)

Esta secção tem como objectivo apresentar os principais aspectos das especificações tecnológicas da norma 802.16 respeitantes à camada MAC. A Figura 7 representa as camadas especificadas na norma 802.16.

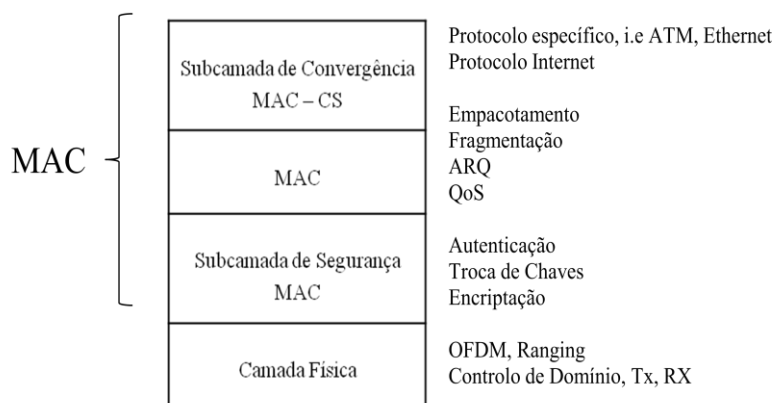


Figura 7 - Camadas da norma IEEE 802.16

A subcamada de Convergência Específica (CS – *Convergence Sublayer*) e a camada MAC integram as partes fundamentais de controlo dos recursos limitados de rádio da interface aérea partilhados pelas BSs e as SSs [6].

2.6.1. CONVERGÊNCIA ESPECÍFICA (CS – *CONVERGENCE SUBLAYER*)

A subcamada de convergência específica (CS) inclui as funcionalidades específicas de adaptação necessárias aos possíveis clientes da rede WIMAX. A norma apresenta duas especificações para a camada CS: a ATM CS e a *Packet CS*.

A ATM CS é uma interface lógica que associa diferentes serviços ATM à subcamada de convergência. A ATM CS foi especificamente definida para dar suporte a convergência dos PDUs (*Protocol Data Units*) gerados pelo protocolo da camada ATM de uma rede ATM. Esta aceita células ATM, classifica, suprime o cabeçalho (PHS – *Payload Header Suppression*) se solicitado e entrega o PDU ao *Service Access Point* (SAP) apropriado.

A *Packet CS* executa duas funções principais:

- Classificação de pacotes;
- Supressão do cabeçalho *payload*.

O processo de classificação de pacotes dá-se quando a subcamada de convergência recebe um pacote de uma camada superior e mapeia este pacote num fluxo de serviço. O fluxo de serviço é definido como uma ligação com um conjunto de parâmetros de QoS. Uma vez que cada fluxo de serviço está associado a parâmetros específicos de QoS, a classificação de um pacote para um fluxo leva a que a entrega deste seja feita de acordo com as restrições dos QoS adequados [6].

A classificação dos pacotes é feita com base em diferentes critérios, tais como endereços IP de origem ou destino. Se um pacote corresponde a um determinado critério, é entregue à ligação que seja correspondente a este mesmo critério, isto é, a classificação resulta num identificador de ligação (CID – *connection ID*) associado a esse critério, como indica a Figura 8.

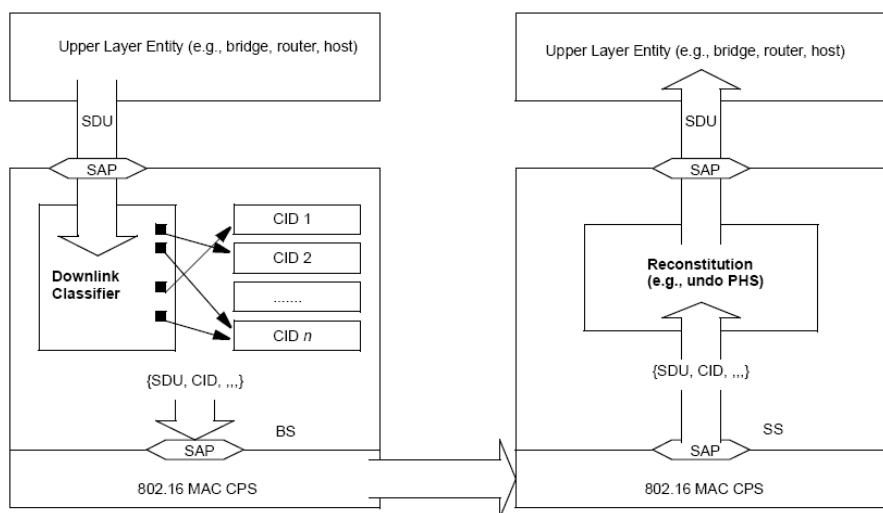


Figura 8 - Classificação e mapeamento CID [6]

Vários classificadores podem existir para o mesmo fluxo de serviço; uma vez que se podem sobrepor, estes são especificamente ordenados.

A capacidade de suprimir cabeçalhos de *payload* (PHS – *Payload Header Suppression*) é opcional sendo usada para remover informação repetida ou redundante de cabeçalhos de camadas superiores. Um índice de repressão do cabeçalho do pacote (PHSI) é adicionado ao pacote como uma referência ao campo PHSF (*Payload Header Suppression Field*) do lado da recepção.

Depois de passar através da subcamada de convergência, o pacote entregue ao fluxo de serviço apropriado na camada MAC tem o formato mostrado na Figura 9. A norma denomina o formato completo de MAC SDU (*MAC Service Data Unit*) que contem o PHSI e o PDU da camada superior.

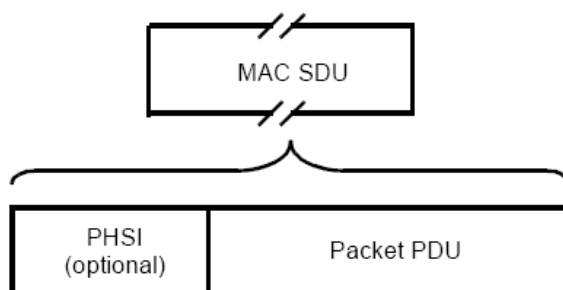


Figura 9 - Formato MAC SDU [6]

O MAC SDU é da responsabilidade da camada MAC sendo esta encarregada pela entrega do pacote ao receptor.

2.6.2. SUBCAMADA MAC

A camada MAC é a camada que controla e distribui os recursos limitados do espectro radioelétrico entre as BSs e as SSs [18], enquanto a camada física lida com funções específicas radioelétricas como modulação, codificação e alocações físicas de frequência. Sendo o meio físico um ambiente wireless que está limitado pela largura de banda radioelétrica, é importante que a distribuição dos recursos seja feita da forma mais eficiente possível.

Ficam aqui expostas algumas das funções da camada MAC que serão aprofundadas no decorrer deste documento:

- Escalonamento de dados;
- Configuração do QoS e a sua manutenção;
- Manutenção das ligações;
- *Handovers*, modos *Idle/sleep*;

As especificações da norma 802.16 definem dois modos de operação: modo PMP ou modo *mesh*. O modo *mesh* é usado principalmente em redes ad hoc ou sistemas similares que possam fazer uso das capacidades das SSs de executar *routing* ou enviar informação directamente entre si. Esta secção abrange a camada MAC no modo PMP, sendo grande parte da teoria aplicável também às operações no modo *mesh*.

A camada MAC da norma 802.16 é orientada à ligação sendo todas as comunicações de dados associadas a uma ligação. Uma ligação juntamente com os parâmetros de QoS constituem um fluxo de serviço, que é um termo fundamental na norma. A qualidade de serviço é mantida através de cinco diferentes classes de QoS, que basicamente são diferentes mecanismos de escalonamento [6].

Framing

A camada MAC oferece suporte para ambas as tramas TDD e FDD, separando, no caso do TDD, as subtramas de *uplink* e de *downlink* no tempo, enquanto no FDD essa separação é feita nas frequências. A Figura 10 mostra os *slots* físicos (PS – *Physical Slots*) que compõem a estrutura de uma trama (*frame*) TDD.

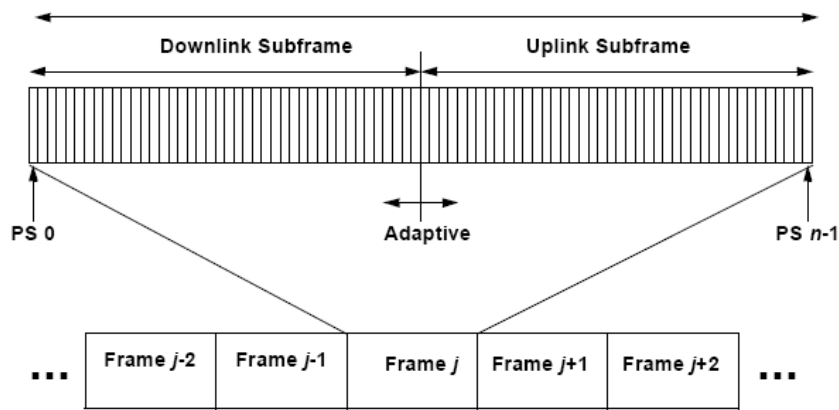


Figura 10 - Estrutura da trama TDD [6]

O tamanho da trama pode ser variável de acordo com perfis físicos diferentes. A partição da trama entre as subtramas de *uplink* e de *downlink* pode também ser ajustada.

Como exemplo do conteúdo das partes de *uplink* e *downlink* de cada trama, a Figura 11 mostra a especificação da estrutura de uma trama OFDM (*Orthogonal Frequency Division Multiplexing*), específico da camada física, que melhor ilustra as relações de PDUs e das partes *up/downlink*. Entre as subtramas de *uplink* e de *downlink* existem espaços de guarda que permitem comutar entre o modo de recepção e de transmissão via rádio. Estes espaços de guarda são chamados de TTG (*Transmit/receive Transition Gap*) e RTG (*Receive/transmit Gap*), respectivamente.

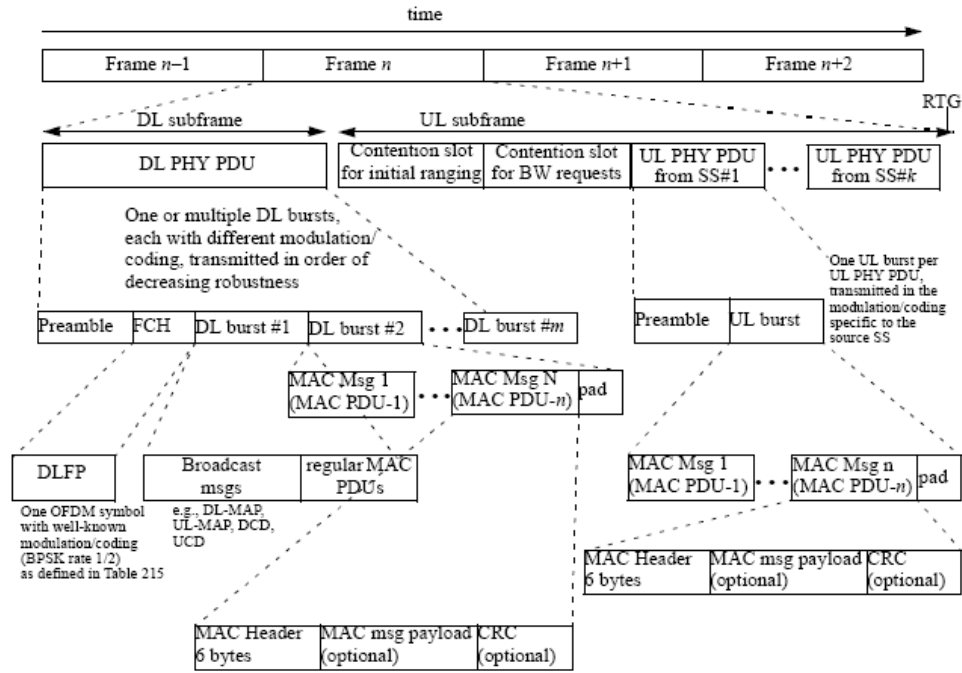


Figura 11 - Estrutura da trama OFDM [6]

Subtrama de Downlink

A subtrama de *downlink* (DL) é constituída por diversas “rajadas” físicas, *bursts*, de diferentes modulações/codificações [6]. Estes *bursts* são endereçados em diferentes ligações através dos identificadores de ligação (CID – *Connection Identifiers*). Os identificadores podem reconhecer *bursts* endereçados para SSs individuais, mensagens de *broadcast* ou *multicast*. Após os preâmbulos de sincronização e outros dados necessários à camada física, ocorre um *burst* de *broadcast*. Este contém importantes mensagens de controlo como mensagens de mapeamento DL-MAP (Downlink-MAP) e UL-MAP (*Uplink*-MAP) e ainda mensagens de descrição do canal UCD (*Uplink Channel Descriptor*) e DCD (*Downlink Channel Descriptor*).

O DL-MAP, se programado para a transmissão na trama, aparece sempre em primeiro lugar descrevendo o conteúdo da subtrama de *downlink*. Contém o identificador da BS bem como do operador contendo também elementos de informação (IE – *Information Elements*) respeitantes aos *bursts* da subcamada de *downlink*, isto é, para a transmissão da BS para diferentes SSs.

Estes IEs no DL-MAP agem como apontadores para os *bursts* da subtrama de *downlink*, uma vez que contêm informação sobre qual o CID que o *burst* tem como destino e qual o

seu tipo. Isto serve para permitir que a SS, através do DL-MAP identifique que *bursts* deve escutar e como deve configurar os seus parâmetros radioelétricos de modo a poder decodificá-los.

A Figura 12 mostra um exemplo de como o DL-MAP contém informação relativa ao momento em que um *burst* se inicia. O DIUC (*Downlink Interval Usage Code*) e o seu homólogo do *Uplink-MAP*, o UIUC (*Uplink Interval Usage Code*) são índices referentes ao tipo do *burst*. Este é o índice que é encontrado nos detalhes das mensagens periódicas UCD e DCD.

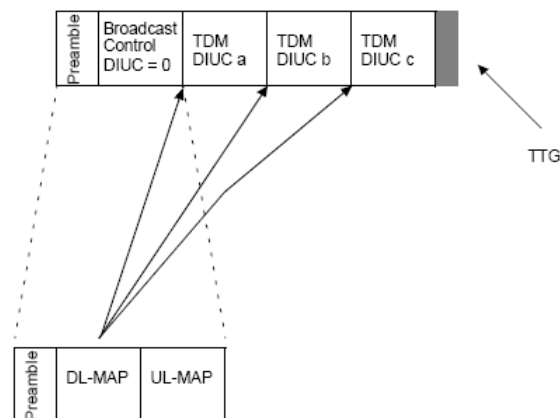


Figura 12 - DL-MAP feitas para *bursts* da subtrama de downlink de vários tipos [6]

O UL-MAP apresenta funções muito idênticas às funções desempenhadas pelo DL-MAP, apenas este caso reporta-se aos *bursts* da subtrama de *uplink*. Estes *bursts* foram alocados pela BS para as SSs enviarem dados via *uplink*. Os IEs no UL-MAP descrevem quais os tipos de *burst* e qual o tempo do início da alocação dos *bursts*.

A Figura 13 demonstra como as mensagens de mapeamento DL-MAP e UL-MAP apontam para as alocações no modo TDD.

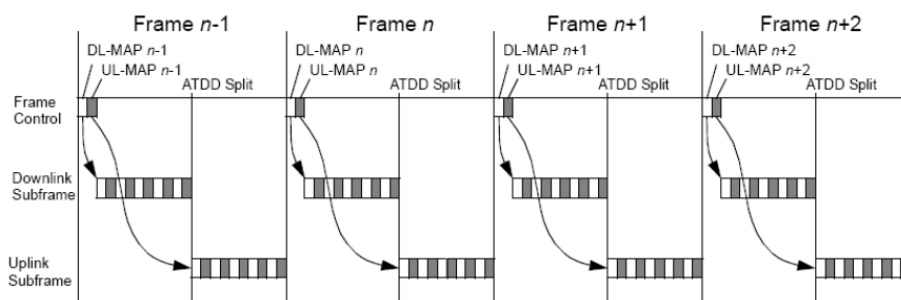


Figura 13 - DL-MAP e UL-MAP em TDD [6]

As mensagens UCD e DCD definem as características detalhadas dos *bursts* das subtramas de *downlink* e de *uplink*. Estas mensagens contêm os detalhes de como deverá ser decodificado cada *burst*. Estas mensagens são transmitidas em intervalos de menor frequência que os UL/DL-MAP. As alterações de modulações de cada tipo de *burst* são assinaladas através do contador UCD/DCD. Um *burst* pode mudar as suas características físicas adaptando-se ao meio ou a requisitos de taxas de dados. Os contadores permitem que as estações ignorem informações mais antigas até que uma alteração seja detectada.

Subtrama de *Uplink*

A subtrama de *uplink* é partilhada pelas SSs onde cada uma usa a sua codificação e modulação para transmitir para a BS. Os elementos de informação (*Information Elements – IEs*) do UL-MAP enviados na subtrama de *downlink* indicam a que SS cada *burst* é endereçado e que tipo de *burst* se trata. Uma parte da subtrama de *uplink* é reservada para realizar *ranging*, que é definido como um método que determina a distância da posição da SS até à posição da BS inicial. Este processo é usado para uma SS atingir sincronização e níveis de sinal apropriados para futuras comunicações com a BS. A outra parte serve para pedidos da largura de banda, feitos também numa forma de contenção, quando uma SS pode enviar pedidos sem ficar à espera de alocações individuais de *bursts*. Estes dois períodos são feitos numa forma de contenção (disputa de pedidos entre SSs), sendo possível ocorrerem colisões entre SS que transmitam ao mesmo tempo. Para combater estas colisões as mensagens UCD contêm informação sob uma janela *Backoff* para ambos os pedidos de intervalos de largura de banda e de *ranging*. Quando uma SS quer transmitir num destes períodos de contenção, configura a sua janela de contenção de acordo com a mensagem UCD.

Nos *bursts* seguintes, cada SS possui as suas próprias porções da subtrama de *uplink* para o envio de dados usando um tipo de *burst* específico.

A Figura 14 representa os procedimentos para a entrada na rede de uma SS.

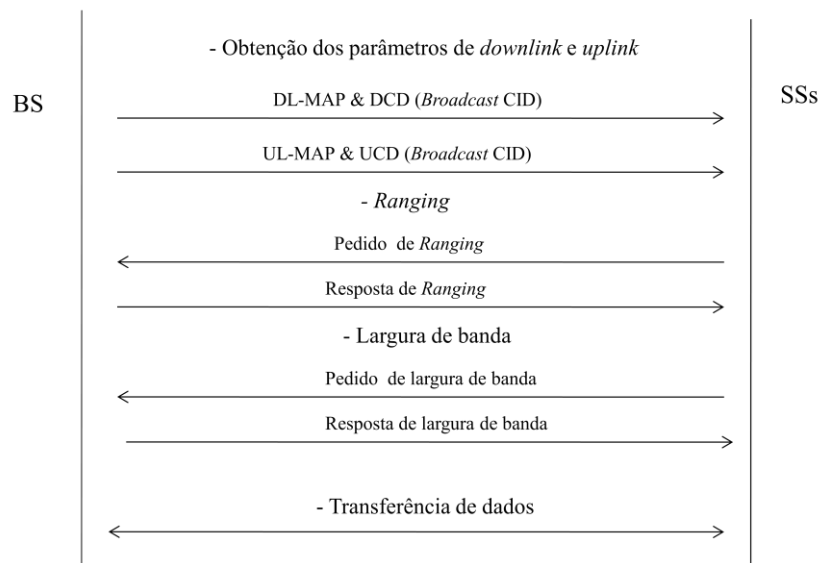


Figura 14 - Procedimento para entrada na rede de uma SS

MAC PDU

A norma 802.16 dispõe de um vasto leque de mensagens de gestão. Estas mensagens e os pacotes de dados da camada superior podem ser empacotados e enviados usando vários métodos. A norma faz uso de um esquema de formato TLV (*Time/Length/Value*, tempo/comprimento/valor) permitindo um conteúdo mais dinâmico das mensagens. Os vários parâmetros são enviados através destes esquemas permitindo uma análise por tipo, comprimento ou valor predefinido. Deste modo as mensagens podem conter um número variável de parâmetros.

Os PDUs podem ser ligados para preencher um *burst* que lhes tenha sido atribuído, ou então fragmentados. A fragmentação permite uma utilização mais eficiente da largura de banda atribuída no que diz respeito aos requisitos de QoS. Existe também suporte para o protocolo ARQ (*Automatic Repeat Request*) permitindo efectuar a retransmissão de mensagens erradas. ARQ é uma técnica de detecção e correcção de erros. O receptor detecta erros numa mensagem recebida, não a podendo corrigir, este envia um pedido de retransmissão.

A Figura 15 mostra o formato do MAC PDU com um tamanho variável dividido em três porções: um cabeçalho genérico de tamanho fixo (6 bytes); um payload de tamanho variável e um código de redundância cíclica (CRC) opcional (4 bytes). O payload pode

encontrar-se vazio ou preenchido com sub-cabeçalhos, MAC SDUs (*Service Data Unit*) ou fragmentos de MAC-SDUs.

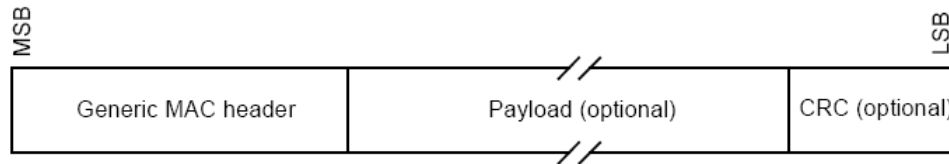


Figura 15 - Formato de um MAC PDU [6]

Ligações

Cada ligação configurada pela BS é identificada por um CID, que funciona como um índice para procurar tipos de serviços ou outros parâmetros. Quando uma SS entra numa célula de uma BS e executa os passos necessários para o início de uma comunicação é-lhe concedida três tipos de ligações de controlo, dois principais e um opcional.

- Ligação básica
- Controlo primário
- Controlo secundário.

Estas ligações estão associadas a níveis de QoS que diferenciam o controlo em três graus de prioridade. A ligação básica é utilizada para alturas críticas, mensagens curtas de gestão MAC, enquanto que a ligação de controlo primário entregue as mensagens MAC mais tolerantes a atrasos. A ligação de controlo secundário é utilizada para protocolos básicos como DHCP, TFTP4, etc. SSs com configurações estáticas não necessitam do uso das ligações de controlo secundárias. Quando estas três ligações de gestão estão criadas, cada uma recebe um CID que identifica a ligação. O CID é o mesmo tanto para a BS como para a SS e é mediado através das mensagens de controlo MAC RNG-RSP e REG-RSP durante a entrada na rede ou na fase de associação à BS.

A norma especifica qual o tipo de mensagens de controlo que usa cada tipo de ligação. Para além destas ligações de controlo também há a possibilidade de configurar as ligações de transporte. Estas ligações guardam os PDUs entregues à camada MAC provenientes de

camadas superiores sendo associadas ao serviço de escalonamento com base nos seus requisitos de QoS.

As ligações representam o nível mais baixo dos serviços de transporte de dados e estão associadas ao nível mais elevado de fluxo de serviço (*service flow*). Como foi visto anteriormente estas ligações, juntamente com os parâmetros de QoS, constituem um tipo de fluxo de serviço.

Escalonamento

Depois de entrar na rede, cada SS cria uma ou mais ligações para que os dados sejam transmitidos de, ou para uma BS. A camada MAC programa o uso dos recursos do meio aéreo fornecendo diferentes classes de QoS (*Quality of Service*) [18]. Cada ligação está associada com um único serviço de dados fornecido por uma classe de QoS, em que cada uma destas classes possui parâmetros que definem os comportamentos destes serviços. Fica aqui exposta uma breve descrição de cada uma destas classes, incluindo também os parâmetros mais relevantes a ter em consideração para cada tipo de serviço que está a ser transmitido [6].

- *Unsolicited Grant Service* (UGS) – Foi projectado para suportar fluxos de dados em tempo real, consistindo em pacotes de dados de tamanho fixo emitidos em intervalos periódicos de tempo, tal como T1/E1 e VoIP (*Voice over IP*) sem repressão de silêncio. Os parâmetros essenciais à programação deste tipo de fluxo de serviço são:
 - i. Taxa de transferência efectiva máxima de tráfego;
 - ii. Latência máxima;
 - iii. *Jitter* tolerado;
 - iv. Política de pedido/transmissão;
 - v. Se presente, o parâmetro da taxa de transferência efectiva mínima de tráfego reservado deve ter o mesmo valor que o parâmetro da taxa de transferência efectiva máxima de tráfego.
- *Real-time-Polling Service* (rtPS) – Foi planeado para sustentar fluxos de dados em tempo real formados em pacotes de dados de tamanho variável, emitidos em intervalos periódicos de tempo, tal como MPEG (*Moving Pictures Experts Group*) vídeos. Os parâmetros principais a ter em conta nesta classe são:
 - i. Taxa de transferência efectiva mínima de tráfego reservado;

- ii. Taxa de transferência efectiva máxima de tráfego;
 - iii. Máxima latência;
 - iv. Política de pedido/transmissão.
- *Non-real-time Polling Service* (nrtPS) – Foi delineado para suportar fluxos de dados com tolerância de atraso consistindo em pacotes de dados de tamanho variável em que uma mínima taxa de tráfego é requerida, tal como o FTP. Os parâmetros fundamentais a ter em conta são:
 - i. Taxa de transferência efectiva mínima de tráfego reservado;
 - ii. Taxa de transferência máxima de tráfego;
 - iii. Prioridades de tráfego;
 - iv. Políticas de pedido/transmissão.
- *Best Effort* (BE) – Foi projectado para suportar fluxo de dados em que não é necessário assegurar um nível mínimo de serviço, logo pode ser assegurado numa base espaço/disponibilidade. Os parâmetros essenciais a ter em conta são:
 - i. Taxa de transferência efectiva máxima de tráfego;
 - ii. Prioridades de tráfego;
 - iii. Políticas de pedido/transmissão.
- *Extended real-time Polling Service* (ertPS) – Este novo serviço foi introduzido com a nova norma IEEE 802.16e [7], sendo projectado para ter a eficiência de ambos os serviços UGS e rtPS. A BS deve suportar todos os serviços disponíveis pela classe UGS salvaguardando a latência de um pedido de largura de banda. A diferença é que no serviço UGS as alocações são de tamanho fixo, enquanto no ertPS as alocações são dinâmicas. Uma aplicação típica para este tipo de serviço é o VoIP.

A Tabela 3 representa as definições de cada classe bem como as suas aplicações.

Tabela 3 - Classes de QoS

Serviço	Abrev.	Definição	Aplicações usuais
Unsolicited Grant Service	UGS	Fluxos de dados em tempo real compreendendo pacotes de dados de dimensão fixa em intervalos periódicos de tempo	Transporte T1/E1
Extended Real-time Polling Service	ErtPS	Fluxos de serviço em tempo real que geram pacotes de dados de dimensão variável em intervalos periódicos	VoIP
Real-time Polling Service	rtPS	Fluxos de dados em tempo real compreendendo pacotes de dados de dimensão variável em intervalos periódicos de tempo	Video MPEG
Non-real-time Polling Service	nrtPS	Fluxos de dados tolerantes a atrasos compreendendo pacotes de dados de dimensão variável que requerem valores mínimos de taxa de transferência	FTP com garantias mínimas de taxa de transferência
Best Effort	BE	Fluxos de dados que não requerem qualquer mínimo nível de serviço	HTTP

Como já foi referido, estes tipos de serviços e os mecanismos para a requisição da largura de banda são definidos, mas não são definidas as formas de escalonamento e de gestão dos mesmos. Fica assim em aberto a implementação de algoritmos de escalonamento, controle de admissão e policiamento de tráfego (Figura 16).

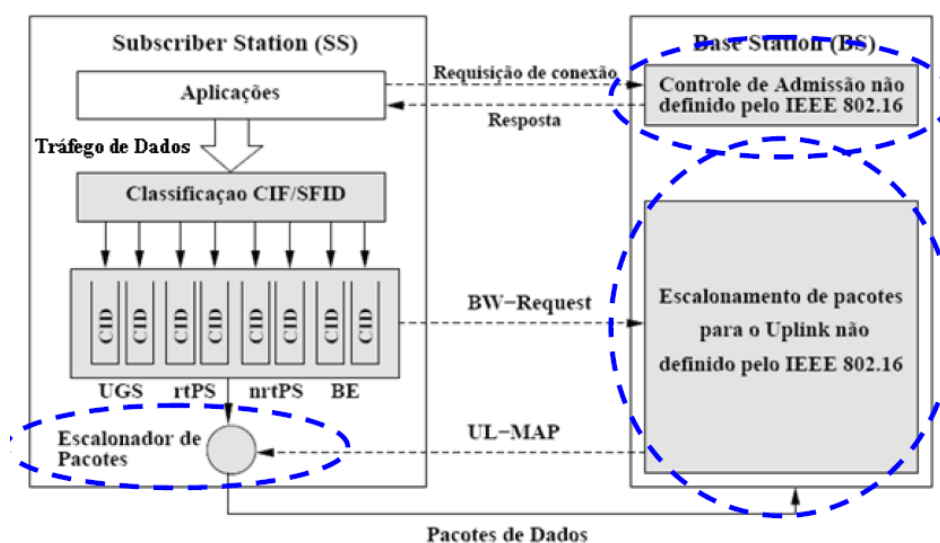


Figura 16 - Arquitectura possível de QoS da norma 802.16 [9]

Requisição de largura de banda

Uma SS pode efectuar um pedido de largura de banda enviando uma mensagem específica para o pedido ou então através de um *Piggyback Request* num subcabeçalho de gestão. Estes pedidos podem ser enviados em qualquer *burst uplink* excepto durante o intervalo inicial de *ranging*.

Uma SS requer sempre largura de banda referenciando-se a uma ligação particular, mas a BS aloca a largura de banda pedida endereçando-a numa ligação de controlo básico e não a uma ligação individual. Isto faz com que a SS desconheça a que pedido se refere a concessão dada. Uma BS é capaz de atribuir a largura de banda especificamente a uma mensagem de pedido de largura de banda efectuada por uma SS. Isto pode ser feito tanto para grupos de SSs como para SSs individuais vindo na forma de *bursts* específicos no UL-MAP [6].

2.7. SUBCAMADA DE SEGURANÇA

As principais funções da subcamada de segurança são:

- Sigilo de informações;
- Autenticação e autorização;
- Certificação digital.

Esta subcamada fornece privacidade aos assinantes da rede wireless através da encriptação das ligações entre a SS e a BS, assim como uma forte protecção contra roubo de serviços. A BS protege contra acessos não autorizados aos serviços de transporte de dados forçando a encriptação dos fluxos de serviço através da rede. Nesta subcamada são empregues um protocolo de encapsulamento, para encriptação dos pacotes de dados (este protocolo também define as criptografias suportadas), análise dos dados encriptados, algoritmos de autenticação, e regras de aplicação destes algoritmos no MAC PDU *payload*. Também é utilizado um protocolo de gestão de chaves (PKM – *Key Management Protocol*). Este protocolo é utilizado pela SS para obter autorização e tráfego dos dados da chave da BS, reautorização periódica e actualização de chave. O PKM utiliza certificação digital X.509, algoritmo de encriptação RSA de chave pública (PKCS #1) e fortes algoritmos de

criptação para actuar na troca de chaves entre a SS e a BS. Este protocolo foi criado através do conceito de *security associations* (SAs), que são um conjunto de métodos de criptografia e dados da chave associados. As SAs contêm as informações sobre quais os algoritmos de encriptação que devem ser utilizados, qual chave utilizar, etc. [6]

2.8. CAMADA FÍSICA

A camada física opera num formato de *frames*, que são organizadas em intervalos de tempo, chamados *slots* físicos. Em cada frame há um *subframe downlink* e um *subframe uplink*. O *subframe downlink* é usado pela BS para a transmissão de dados e de informações de controlo para as SSs. O *subframe uplink* é compartilhado entre todas as SSs para transmissões que têm como destino a BS.

As principais funções da camada física são:

- Transmissão dos MAC PDUs;
- Definição das técnicas de transmissão digital: modulação e codificação;
- Definição do espectro;
- Correção de erro;
- Construção dos *frames* e *sub-frames* de transmissão.

Seguidamente serão apresentados, em termos gerais, diferentes especificações da camada física que a norma IEEE 802.16 padroniza.

2.8.1. WIRELESS METROPOLITAN AREA NETWORK – SINGLE CARRIER

O *Wireless Metropolitan Area Network – Single Carrier* (WirelessMan-SC) [6], opera na faixa de 10-66GHz. Suporta TDD e FDD. Permite utilizar vários perfis de transmissão adaptativos (ABPs – *Adaptive Burst Profiles*), nos quais os parâmetros de transmissão podem ser ajustados individualmente, para cada estação. O *uplink* é baseado numa combinação de TDMA e DAMA (*Demand Assigned Multiple Access*). O *downlink* é TDM, fazendo o *broadcast* da informação destinada às estações de um mesmo sector. FDD suporta *fullduplex* e *half-duplex*, porque utiliza duas frequências distintas podendo usar modulações diferentes. No TDD o *frame* possui um tamanho fixo, sendo possível ajustar-

se à porção do *subframe* destinada a *downlink* (*Downlink Subframe*) e a *uplink* (*Uplink Subframe*). Desta forma, é possível ajustar-se a capacidade de transmissão em ambas as direcções. Em cada *frame*, os *slots* de tempo (TDM DIUC – *Downlink Interval Usage Code*) são alocados em grupo para uma ou mais SSs. Mais precisamente, estes *slots* de tempo são reservados para ligações pertencentes a uma ou mais SSs. Um mapeamento de *downlink* é enviado pela BS no início do *subframe* para identificar onde estão os *slots* de tempo destinados a uma determinada SS. Este mapeamento detalha quando é que os dados estão a ser esperados, para uma determinada ligação, bem como os parâmetros de transmissão utilizados (ABP). A alocação dos *timeslots* no *downlink* é feita pela BS. Para que o TDD funcione, é necessário a sincronização de frames na BS e SSs. Ou seja, SS *frames* devem estar sincronizados com os *frames* da BS. Este é o maior problema do TDD, uma vez que o atraso de propagação faz com que os quadros sejam recebidos após um intervalo de tempo no destino. Para resolver este problema, é necessário que a SS inicie a transmissão dos seus *slots* de tempo no *subframe* de *uplink*, antecipadamente.

2.8.2. **WIRELESS METROPOLITAN AREA NETWORK – SINGLE CARRIER(A)**

O *Wireless Metropolitan Area Network – Single Carrier(a)* (WirelessMAN-SCa) [6] é baseada em transmissão de portadora simples (SC – *Single Carrier*), projectada para sistemas sem linha de visão directa (NLOS) e para operar na faixa de sub 11 GHz. Suporta TDD e FDD. *Downlink* é TDM ou TDMA. *Uplink* é TDMA. Suporta ABP e FEC, tanto no *uplink*, como no *downlink*. Acrescenta melhorias na estrutura dos quadros visando contornar as condições do meio de transmissão NLOS.

2.8.3. **WIRELESS METROPOLITAN AREA NETWORK - ORTOGHONAL FREQUENCY DIVISION MULTIPLEXING**

O *Wireless Metropolitan Area Network – Orthogonal Frequency Division Multiplexing* (WirelessMAN-OFDM) [6] é baseado na modulação OFDM, projectada para sistemas sem linha de visão directa (NLOS) e para operar na faixa de sub 11 GHz. Possui 256 subportadoras no total, destas, somente 200 transportam dados. Possui 55 portadoras de guarda. Suporta TDD e FDD. Suporta ABP e FEC, tanto no *uplink*, como no *downlink*. Um *frame* também consiste num *Downlink Subframe* e um *Uplink Subframe*. Um *Downlink Subframe* consiste num único PHY PDU. Um *Uplink Subframe* consiste num ou mais PHY PDUs. O PHY PDU inicia com um grande preâmbulo, que é utilizado para fins de sincronização. Após este preâmbulo, existe um campo de controlo chamado FCH (*Frame*

Control Header) que serve para diversos propósitos, incluindo mapeamentos. Depois do FCH existem vários *Downlink Bursts*, cada qual podendo utilizar diferentes ABPs. Esta camada possui estruturas de transmissão diferentes, dependendo se a topologia é PMP (*Point-Multipoint*) ou *Mesh*.

2.8.4. **WIRELESS METROPOLITAN AREA NETWORK - ORTOGHONAL FREQUENCY DIVISION MULTIPLE ACCESS**

O *Wireless Metropolitan Area Network - Orthogonal Frequency Division Multiple Access* (WirelessMAN – OFDMA) [6] utiliza OFDM com 2048 subportadoras. Uma SS pode utilizar mais que uma subportadora, daí o termo *Multiple Access*. A utilização de 2048 subportadoras torna a FFT mais lenta e aumenta os requisitos de sincronização. Por estes e outros motivos, este sistema actualmente tem despertado menos interesse da indústria do que o de 256 subportadoras.

No OFDM, todas as portadoras são transmitidas em paralelo com a mesma amplitude. O OFDMA divide o espaço de portadora (*carrier space*) em N_G grupos, em que cada um tem N_E portadoras, e em N_E sub-canais, cada um com uma portadora por grupo (Figura 17). Codificação, modulação e amplitude são regulados separadamente em cada sub-canal baseando-se nas condições do canal para otimizar o uso dos recursos da rede [8].

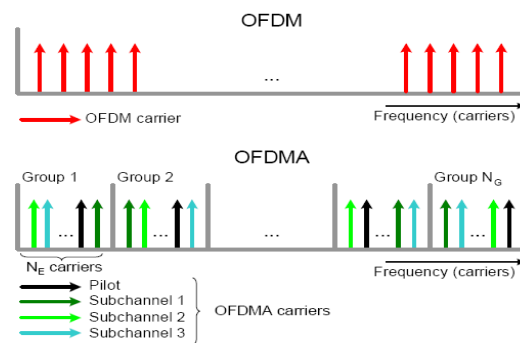


Figura 17 - OFDM e OFDMA [8]

2.8.5. **WIRELESS HIGH-SPEED UNLICENCED METROPOLITAN AREA NETWORK**

Para faixas de frequências não licenciadas, uma outra especificação de camada física foi proposta, *Wireless High-Speed Unlicensed Metropolitan Area Network* (WirelessHUMAN) [6], prevendo a operação em conjunto com outros dispositivos na mesma faixa de frequência. Esta camada física é mais um conceito do que uma

implementação em si, trata-se de um sistema com 5 MHz de espaçamento entre canais a operar de 5 GHz até 6 GHz.

2.9. RESUMO

O segundo capítulo introduz a tecnologia WIMAX que tem como propósito fornecer um acesso de banda larga *wireless* capaz de abranger uma grande área de cobertura podendo atingir um alcance até 50 km e uma taxa de transferência até 70Mbps. O WIMAX enquadra-se em redes da escala MAN sendo capaz de oferecer Qualidade de Serviços (QoS) aos utilizadores.

É exposto a evolução da norma IEEE 802.16, norma basilar do WIMAX, apresentando a organização WIMAX Fórum, que acompanha o desenvolvimento desta tecnologia nas suas diversas vertentes, assegurando a interoperabilidade e conformidade entre equipamentos baseados na norma IEEE 802.16.

Neste capítulo é apresentada a arquitectura de referência desta tecnologia onde se destacam entidades como a estação base, BS, e as entidades dos assinantes, SSs ou MSs. O WIMAX fornece dois tipos de acesso: um acesso fixo wireless suportando três topologias arquitectónicas, ponto-a-ponto, ponto-a-multiponto e multiponto-a-multiponto (mesh), e um acesso móvel onde é feita a referência às duas questões essenciais para aplicações moveis: *handover* e tempo de vida da bateria.

É ainda introduzida uma exaustiva descrição dos principais aspectos das especificações tecnológicas da norma 802.16 respeitantes à camada MAC onde se destacam as subcamadas CS, MAC e a subcamada de segurança. É descrito também como é realizado o escalonamento dos mecanismos de controlo de qualidade de serviço apresentando as diferentes classes de QoS (UGS, ertPS, rtPS, nrtPS e BE).

Por fim, é feita uma breve descrição da camada física apresentando as diferentes especificações que a norma 802.16 padroniza.

3. INVESTIGAÇÃO DE QoS EM WIMAX

Este capítulo apresenta alguns dos trabalhos de investigação dos aspectos de QoS no seio da comunidade científica. A investigação de mecanismos de gestão da qualidade de serviço em WIMAX apresenta-se como um dos temas fulcrais para o bom funcionamento desta tecnologia. Assim, são vários os trabalhos científicos apresentados que propõem soluções de controlo de QoS. Os trabalhos expostos neste capítulo não são disponibilizados para simulação de forma a serem testados. Foi feita uma investigação teórica, nesta área, reunindo os trabalhos que apresentam as soluções mais interessantes.

3.1. “*ENSURING THE QoS REQUIREMENTS IN 802.16 SCHEDULING*” DE ALEXANDER SAYENKO ET AL

Neste artigo [19] são propostos alguns algoritmos para suportar os QoS na camada MAC. A BS é capaz de alocar *slots* baseando-se nos requisitos dos QoS, no tamanho dos pedidos de largura de banda e nos parâmetros da rede WIMAX.

Sendo o WIMAX orientado às ligações, cada SS deve-se registar à BS antes de começar a enviar ou a receber dados. Durante este processo de registo a SS pode negociar o serviço pedido de QoS com a BS. Este pedido pode ser alterado mais tarde e uma nova ligação pode também ser estabelecida. Para garantir o fornecimento de QoS na rede WIMAX é a BS que faz a gestão em ambos os sentidos (*uplink* e *downlink*), ou seja, um algoritmo na BS tem que traduzir o QoS pedido pelas SSs no número de *slots* apropriado. Este algoritmo pode também ser responsável pelo pedido do tamanho da largura de banda que especifica o tamanho do buffer de entrada da SS (este pedido pode ser feito por mensagens independentes ou por *piggy-packed* junto com outros pacotes). Quando a BS faz a sua decisão de escalonamento informa todas a SSs usando as mensagens UL-MAP e DL-MAP no início de cada trama. Esta política de gestão, ou seja, um algoritmo de alocação de *slots* não está definido nas especificações do WIMAX, sendo por isso proposto aqui o seguinte algoritmo:

Várias pesquisas propõem uma complexa forma de escalonamento como uma hierarquia de escalonamento, tal como *Deadline First* (DF), *Deficit Round Robin* (DRR), *Weighted Fair Queueing* (WFQ) e *Worst-case Fair Queueing* (W²FQ). Contudo, este tipo de escalonamento é uma tarefa complicada porque os pedidos de QoS por ligação devem ser traduzidos na configuração de escalonamento em cada nível. Além disso, à medida que a SS envia dados, os tamanhos pedidos variam várias vezes, tendo a BS que atribuir novamente *slots*. Por tudo isto, este artigo propõe o uso de um único nível com um simples mecanismo de escalonamento baseado conceptualmente numa aproximação por *Round-Robin* (RR). Uma solução simples como esta é uma boa solução porque existe muito pouco tempo para se efectuar uma decisão de escalonamento. Por exemplo, um dos valores possíveis de configuração é 400 tramas por segundo, o que significa que cada BS deverá fazer 400 decisões de escalonamento por segundo para alcançar a alocação exacta e justa.

A proposta deste mecanismo de gestão é conceptualmente similar ao *Weighted Round Robin* (WRR) em que cada ligação pode ser tratada por uma sessão separada, enquanto que

o número de *slots*, que a BS tem que alocar para cada ligação baseada no seu pedido de QoS, é o valor do peso do mecanismo WRR. Como todas as *slots* são do mesmo tamanho não é necessário outro tipo de mecanismos mais complexos como o *Fair Queueing* (FQ) ou o DRR. Existem desajustes entre o mecanismo WRR e o funcionamento em ambiente WIMAX. O mais importante é que o WRR tem um comportamento do estilo trabalho-conservador “saltando” as filas vazias e começando a servir a próxima fila quando todos os pacotes da fila corrente são enviados. Como as tramas do WIMAX são de tamanho fixo, o seu comportamento é trabalho-não-conservador. Por outras palavras, se existirem *slots* que não estão a ser alocados a qualquer ligação, então alguns recursos de largura de banda serão desperdiçados. Outro aspecto importante é que em WRR é possível especificar o número de pacotes a enviar de cada fila durante um *round* não podendo influenciar a ordem. No WIMAX existe uma possibilidade de atribuir cada *slot* a uma ligação, especificando assim a ordem. Apesar de este mecanismo ser mais complexo no WIMAX, vai permitir o controlo dos valores de máximo atraso e *jitter*.

Baseando-se nas considerações acima expostas os autores do artigo propuseram que o mecanismo de gestão compreenda 3 etapas principais:

1. Alocação do mínimo número de *slots* – Nesta etapa, a tarefa da BS é calcular o mínimo número de *slots* para cada ligação de forma a garantir os requisitos básicos de QoS.
2. Alocação de *slots* não usados – Nesta etapa, a BS deve atribuir *slots* livres a algumas ligações para evitar um comportamento trabalho-não-conservador.
3. Ordenação dos *slots* – Nesta etapa, a BS deve seleccionar a ordem dos *slots* para melhorar o fornecimento das garantias de QoS.

Com o intuito de avaliar a abordagem proposta realizaram-se simulações em diferentes cenários de utilização.

A Figura 18 representa o 1º Cenário simulado que consiste numa BS com 15 SSs com ligações do tipo BE. A BS está também, ligada por cabo a um nó de destino.

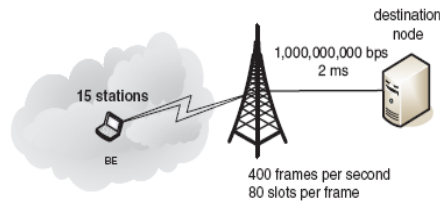


Figura 18 - Estrutura da rede (cenário 1) [19]

Para este cenário, cada SS possui uma aplicação FTP que envia os dados através do protocolo TCP. Para estudar a alocação de recursos com diferente número de SSs activas, foram ligadas ou desligadas as aplicações FTP ao longo do tempo de simulação. No início da simulação, apenas as aplicações das SSs 1,2 e 3 estão activas. A partir dos 5 segundos, mais duas SSs (SS4 e SS5) ficam activas. Aos 10 segundos todas as 15 SSs presentes na rede possuem as suas aplicações ligados. A partir dos 15s apenas duas SSs (SS14 e SS15) permanecem ligadas. Procurou-se aferir a variação da taxa de transferência efectiva (*throughput*) de cada estação ao longo do tempo. A Figura 19 ilustra os resultados medidos pelos autores.

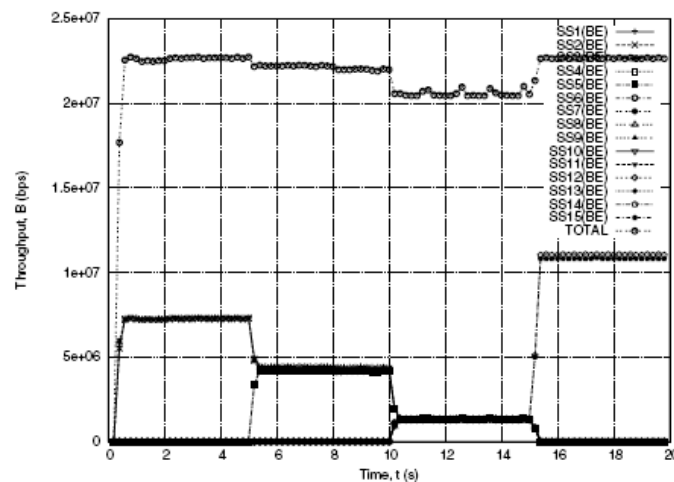


Figura 19 - Resultados obtidos – Throughput – taxa de transferência (cenário 1) [19]

É possível concluir que a BS é capaz de alocar, sempre, os recursos de uma forma justa a todas as SSs. À medida que o tamanho dos pedidos se alteram, o escalonador recalcula o número de *slots* para cada ligação, proporcionando, assim, o mesmo nível de taxa de transferência a cada SS.

Num 2º Cenário alteraram-se as características das ligações entre as diversas SSs e a BS. O cenário representado é composto por uma BS, 1 SS com aplicação UGS, 1 SS com

aplicação ertPS, 2 SSs com aplicações rtPS, 2 SSs com aplicações nrtPS e ainda uma SS com aplicação BE (Figura 20).

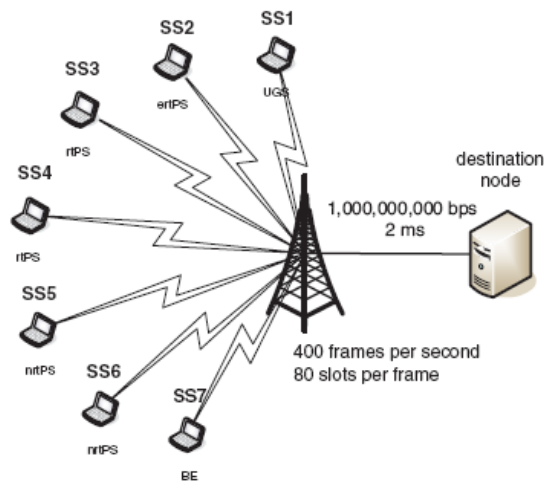


Figura 20 - Estrutura da rede (cenário 2) [19]

Também para este cenário, foram estudadas a taxa de transferência efectiva nas ligações das diversas estações com a BS. Os valores medidos são os mostrados no gráfico da Figura 21.

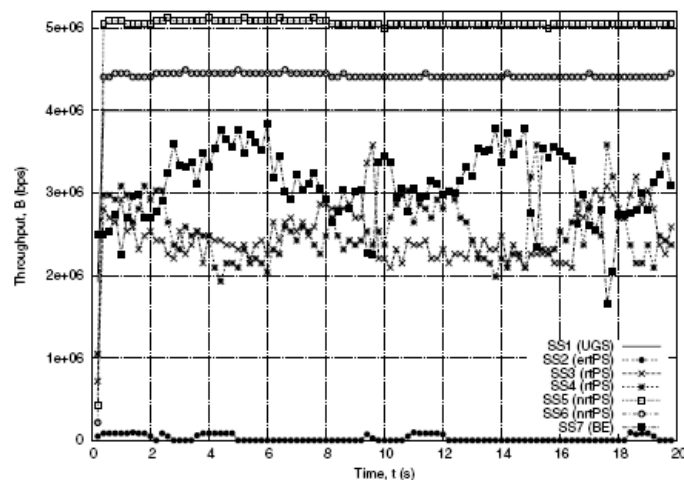


Figura 21 - Resultados obtidos – Throughput – taxa de transferência (cenário 2) [19]

Analisando o gráfico, verifica-se que todas as ligações possuem as taxas de transferências necessárias para o seu bom funcionamento. Os resultados encontrados encontram-se dentro dos limites impostos pelos autores descritos na Tabela 4.

Tabela 4 - Limites da taxa de taxas de transferências impostas [19]

SS	Traffic type	P. size (B)	BW (bps)	
			min	max
SS1	UDP/CBR	500	4,000,000	4,000,000
SS2	UDP/VBR	300	80,000	80,000
SS3	UDP/VBR	1378	2,054,400	4,108,800
SS4	UDP/VBR	1378	2,054,400	4,108,800
SS5	TCP/FTP	1060	3,500,000	5,200,000
SS6	TCP/FTP	1060	3,524,000	4,800,000
SS7	UDP/CBR	200	—	—

Finalmente, num 3º cenário definido por este grupo de trabalho, é considerada a simulação de uma rede composta por 1 BS rodeada por 5 SS com aplicações UGS e ainda uma SS com uma aplicação BE (Figura 22).

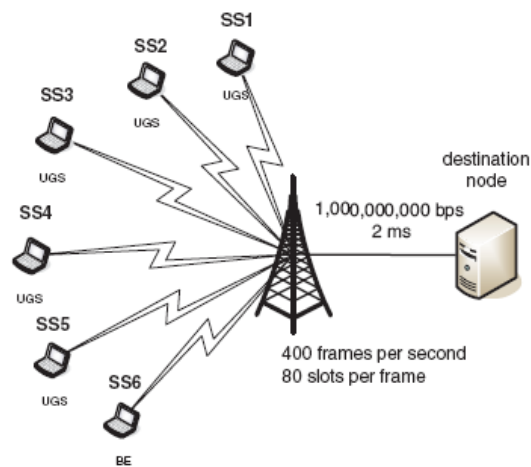


Figura 22 - Estrutura da rede (cenário 3) [19]

Neste cenário, cada SS utiliza diferentes modulações como é representado na Tabela 5.

Tabela 5 - Modulações utilizadas [19]

SS	Traffic type	P. size (B)	Min/Max BW (bps)
SS1	UDP/CBR	100	1,000,000
SS2	UDP/CBR	100	1,000,000
SS3	UDP/CBR	100	1,000,000
SS4	UDP/CBR	100	1,000,000
SS5	UDP/CBR	100	1,000,000
SS6	TCP/FTP	1060	—

Neste contexto, os autores repetem a análise das variações das taxas de transferência efectiva para as diversas ligações, tendo chegado ao conjunto de valores ilustrados no gráfico da Figura 23.

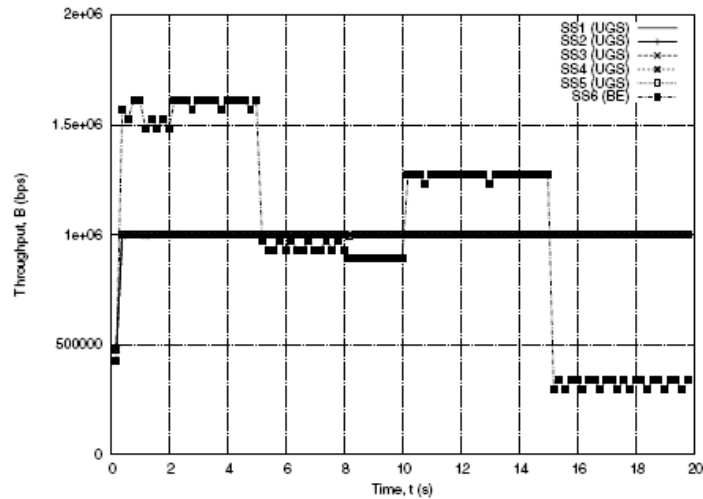


Figura 23 - Resultados obtidos – *Throughput* – taxa de transferência (cenário 3) [19]

Independentemente das modulações utilizadas, verifica-se que cada ligação UGS é sempre provida pelo mesmo número de *slots*. A taxa de transferência efectiva para cada ligação UGS é de 1Mbps. A taxa encontrada para a ligação BE é explicada pela modulação utilizada pelas ligações UGS em certos períodos do tempo de simulação.

3.2. “*PERFORMANCE ANALYSIS OF THE IEEE 802.16 WIRELESS METROPOLITAN AREA NETWORK*” DE DONG-HOON CHO ET AL

Este artigo [20] propõe uma arquitectura que fornece suporte de QoS para tráfego em tempo real (rtPS) com alta prioridade, enquanto mantém o desempenho em níveis aceitáveis para o tráfego de baixa prioridade. Por faltarem partes essenciais na forma de gestão das classes de QoS pela norma 802.16 este artigo propõe algumas alterações na própria arquitectura. Assim, os autores propõem que na BS se acrescente uma parte onde é efectuado o escalonamento para a alocação da largura de banda (algoritmo de escalonamento que suporta todos os tipos de serviços transmitidos) e uma parte de controlo de admissão como é ilustrado na Figura 24. Na SS adicionou-se um módulo para a gestão do tráfego. Para cada tipo de serviço (UGS, rtPS, nrtPS, BE), múltiplas ligações são agregadas aos seus respectivos serviços. Este mecanismo de gestão divide-se em duas etapas. A primeira é efectuada na BS de acordo com a informação do pedido vindo da SS. Assim, o escalonador do *uplink* da SS é responsável pela selecção dos pacotes apropriados de todas as filas e envia-os pelos *slots* de dados de *uplink* concedidos pelo módulo de alocação de pacotes da BS. A BS deve garantir concessões de tamanho fixo de dados a intervalos periódicos para o serviço UGS.

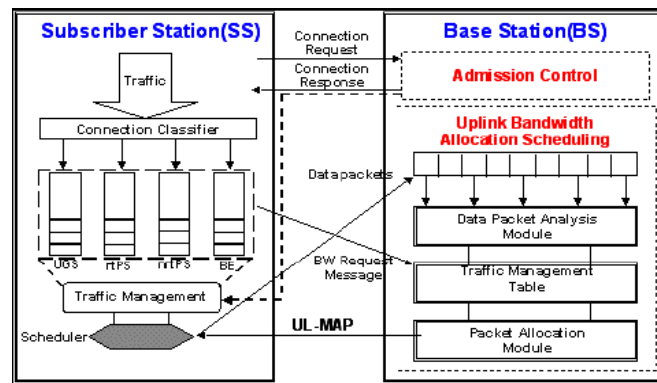


Figura 24 - Arquitetura de QoS proposta [20]

Segundo os autores, o processo de estabelecimento da ligação pode ser sumariado nos seguintes passos [9]:

1. Uma aplicação oriunda de uma SS estabelece uma ligação com a BS usando sinalização de ligação. A aplicação inclui no pedido de ligação o contracto de tráfego (largura de banda e atraso requeridos).
2. O módulo de controlo de admissão na BS aceita ou rejeita a nova ligação.
3. Se o controlo de admissão aceitar a nova ligação, deverá notificar o módulo do escalonador de alocação de largura da banda na BS e fornecer os parâmetros necessários para o módulo de gestão de tráfego da SS.

Depois da ligação estar estabelecida seguem-se os próximos passos:

1. O gestor de tráfego reforça o tráfego baseado no que foi contratado pela ligação.
2. No início de cada trama de tempo, o módulo de análise dos pacotes de dados recolha a informação do tamanho da fila do *BW-request* recebido durante o período da trama anterior. Este módulo irá processar a informação do tamanho da fila e fazer um *update* na tabela de gestão de tráfego.
3. O módulo de alocação dos pacotes retribui a informação do módulo de gestão de tráfego e gera a UL-MAP.
4. A BS faz um *broadcast* da UL-MAP para todas as SSs da subtrama de *downlink*.
5. O escalonador da SS transmite pacotes de acordo com a UL-MAP recebida pela BS.

O escalonador de *uplink* da SS transmite pacotes UGS para verificar o tempo de chegada do pacote virtual do serviço rtPS. Depois da BS receber a informação vinda da SS, vai determinar o *deadline* do pacote, que é o seu tempo de chegada mais a condição de atraso máximo da ligação. Com esta informação a BS pode determinar a ordem do *Poll*, ou seja, a ordem indicada a cada SS para poder transmitir. O serviço rtPS usa o modo livre de contenção ou modo de contenção. Se a SS recebe *Poll* pela BS e não tem uma ligação rtPS na fila, a ligação nrtPS pode transmitir uma requisição *BW-Request*. No caso do serviço BE, este utiliza apenas o modo de contenção.

O escalonador de alocação de largura de banda do *uplink* na BS, com base na informação de *deadline* e usando o mecanismo de *Polling*, escalona as SSs no próximo quadro. A BS então é capaz de conceder o *Poll* para a SS com o *deadline* mais próximo.

Para avaliar a performance do sistema proposto, os autores definiram um cenário composto por uma BS e n SSs. Nas simulações efectuadas foi estudada a ocupação dos diversos canais estabelecidas.

A Figura 25 demonstra a utilização do canal quando é assumido que não existe limite da largura de banda para a classe de maior prioridade (*class 1*).

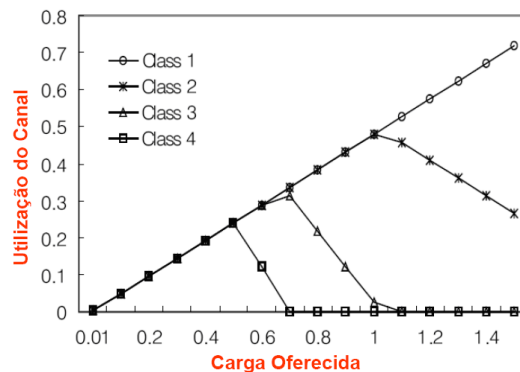


Figura 25 - Utilização do canal quando não existe limite de largura de banda para o tráfego de maior prioridade [9]

Verifica-se que à medida que a carga oferecida na rede vai aumentando, as classes de menor prioridade vão reduzindo a sua utilização do canal.

A Figura 26 representa os resultados encontrados pelos autores quando é permitida, ao fluxo UGS, uma cota fixa de largura de banda.

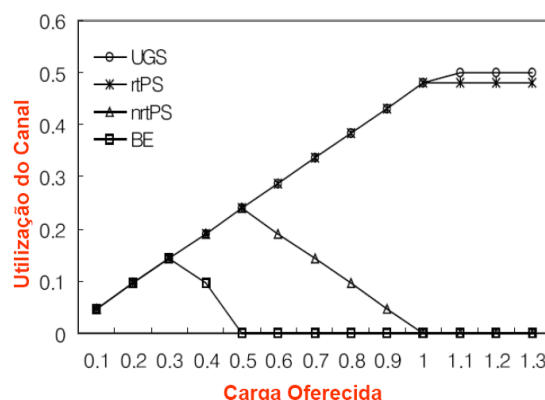


Figura 26 - Utilização do canal quando uma cota fixa da largura de banda é permitida ao fluxo UGS [9]

Os resultados encontrados expõem, tal como na figura anterior, a boa utilização do canal realizada pelo trabalho apresentado.

3.3. “AN INTEGRATED QoS CONTROL ARCHITECTURE FOR IEEE 802.16 BROADBAND WIRELESS ACCESS SYSTEMS” DE JIANFENG CHEN ET AL

Neste [21] trabalho científico é proposta uma abordagem a dois níveis. Uma arquitectura de serviços integrados (IntServ) e uma arquitectura de serviços diferenciais (DiffServ). O IntServ é implementado por quatro componentes: um protocolo de sinalização (RSVP), um controlo de admissão, um escalonador de classificação e um escalonador de pacotes. Além disso, algumas regras são prescritas para classificar pacotes IP DiffServ em diferentes filas com prioridade baseado na indicação de QoS que se encontra no cabeçalho IP. A arquitectura de QoS do modo PMP na norma 802.16-2004 pode suportar tanto IntServ como DiffServ. Para garantir um suporte de QoS num modo rede (*mesh mode*) os autores propõem novas mensagens para a criação (DSA – *Dynamic Service Addition*), mudança (DSC – *Dynamic Service Change*) e eliminação (DSD – *Dynamic Service Deletion*) de fluxos de serviços que garantam os requisitos de QoS *end-to-end*. Um escalonador centralizado é desenvolvido para alocar recursos de rede para o tráfego.

Vários artigos propõe algoritmos que se concentram nos problemas de QoS da camada MAC, mas pouco são aqueles que apresentam uma solução que garanta os requisitos de serviços de QoS para a camada superior. Neste artigo é proposta uma forma de garantir um controlo de *cross layer* QoS.

Na forma tradicional é usada uma sinalização RSVP para efectuar esse controlo. A RSVP pode ser classificada numa fila especial de alta prioridade e ser transmitida na segunda ligação de gestão. Este procedimento resume-se em dois passos. No primeiro, a segunda ligação de gestão será partilhada por RSVP para fornecer o suporte de QoS para o *up-layer* entre muitas mensagens IP relacionadas. No segundo passo, a primeira ligação de gestão irá ser utilizadas pelas mensagens DAS/DSC/DSD para fornecer os QoS da camada MAC.

Uma segunda forma é o caminho proposto pelos autores. Como existem varias semelhanças entre os processos de RSVP na camada IP e o DAS/DSC/DSD na camada MAC, naturalmente, os mecanismos entre estas duas camadas serão superiores em termos de eficácia e rapidez porque da forma tradicional o processo é contornado.

As aplicações devem estabelecer a ligação com a BS antes de serem transmitidos dados uma vez que a comunicação é orientada às ligações. A BS atribui a ligação com um único ID de ligação (CID) a cada transmissão de *uplink* ou *downlink*. A troca de mensagens de DAS e DSC podem ser desenvolvidas para transportar os parâmetros de serviço QoS IntServ. Para serviços DiffServ, por outro lado, um número de comportamentos por salto (*per-hop behaviors* PHBs) de diferentes classes de tráfego podem ser mapeadas directamente em diferentes ligações. Segundo os autores, os requisitos para a arquitectura de controlo de QoS integrados ao nível de multi-camadas devem incluir:

1. Garantia de diferentes níveis de QoS.
2. Prioridades das classes de tráfego.
3. Condução do tráfego de uma forma eficiente.
4. Ajuste da alocação dos recursos dinamicamente.
5. Partilha dos recursos de uma forma justa.

Para alcançar estes requisitos os autores propõem a seguinte arquitectura que implementa um mecanismo *cross layer* de uma forma compreensiva.

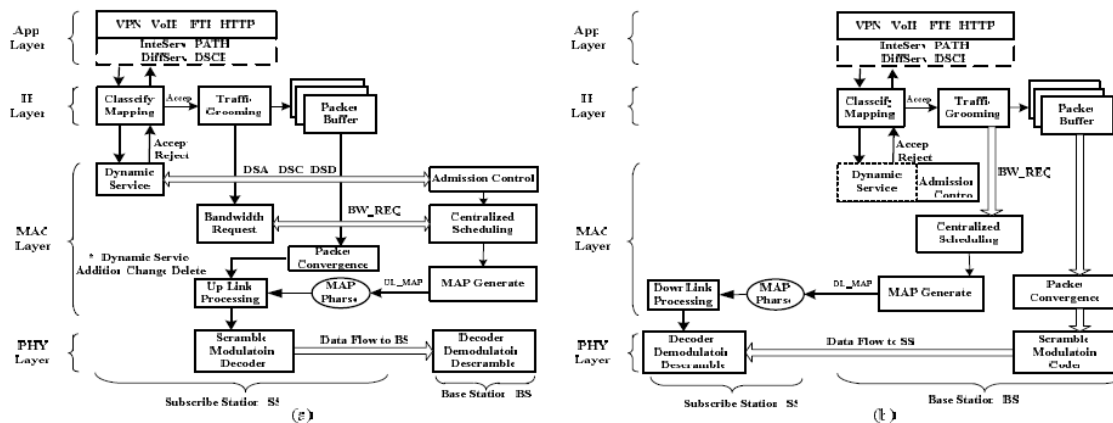


Figura 27 - Arquitectura multi-layer de controlo de QoS integrados [21]

A Figura 27-a ilustra a arquitectura para o tráfego *uplink*. Quando tráfego de uma nova aplicação chega à camada IP, este irá ser analisado de acordo com as definições na *PATH message* (para IntServ) ou *Differentiated Services Code Point* (DSCP para DiffServ). Depois classifica-os num dos quatro tipos de serviço (UGS, rtPS, nrtPS ou BE). No passo seguinte, no módulo *Dynamic Service* na SS irá enviar uma mensagem de pedido para a BS que será aprovada ou não. Se não, o módulo de controlo de admissão irá informar a camada superior para negar o tráfego. Se for aprovada, o módulo de controlo de admissão irá notificar o módulo de escalonamento para garantir os parâmetros necessários de acordo com o valor enviado pela mensagem de pedido e, ao mesmo tempo, o serviço aceite irá ser transferido para o módulo *traffic grooming*. De acordo com o resultado, a SS irá enviar um pedido de largura de banda à BS. O módulo centralizado de escalonamento irá responder ao pedido gerando uma mensagem UL-MAP carregando o resultado da alocação da largura de banda.

A Figura 27-b ilustra a arquitectura para o tráfego *downlink* em que o processamento é praticamente igual ao do *uplink*. A diferença reside no facto das mensagens de pedidos de largura de banda e serviços dinâmicos serem gerados e consumidos na BS. Outra diferença é que a SS apenas necessita de receber dados pelo canal físico de acordo com as instruções da mensagem DL-MAP.

Com o intuito de avaliar a performance da arquitectura proposta no que respeita aos tempos de estabelecimento das ligações e às taxas de transmissão efectiva (*throughput*), foi definido um cenário composto por uma BS e três SSs que operam no modo PMP. Nele foram realizados testes que conduziram aos valores ilustrados na Figura 28 e Figura 29.

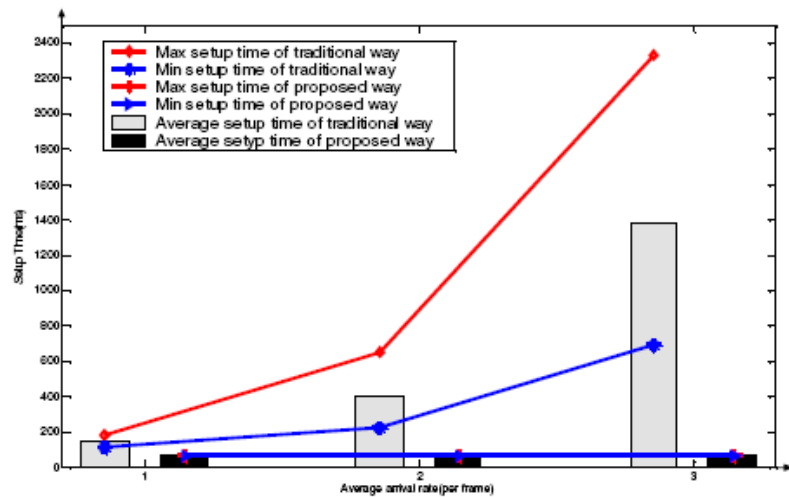


Figura 28 - Tempo Setup vs taxa de chegada [21]

Analisando o gráfico é possível verificar que, com o aumento da taxa chegada de outras mensagens de sinalização, o tempo de setup da forma tradicional aumenta, enquanto que o tempo de simulação da forma proposta não se altera.

A Figura 29 apresenta o mesmo ambiente de simulação descrito, apresentando dois tipos de escalonamento de classes de QoS, *Priority Queue* (PQ) e *Deficit Fair Priority Queue* (DFPQ).

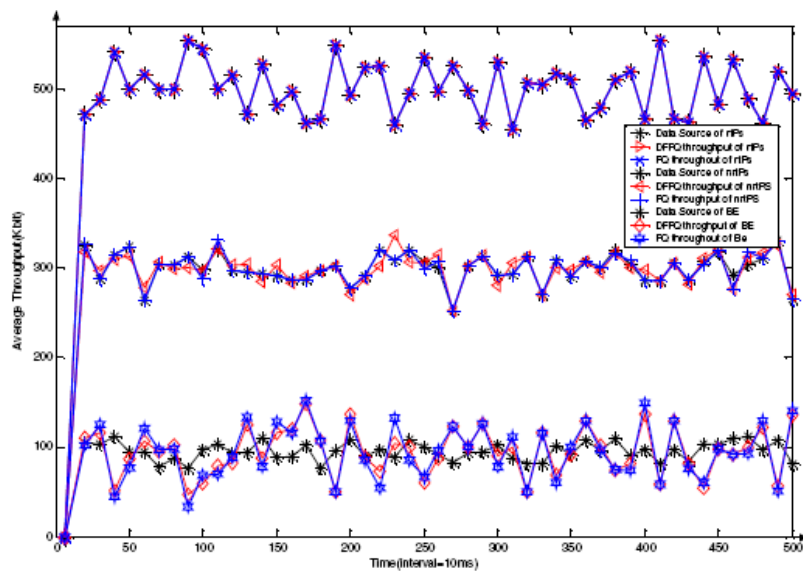


Figura 29 - Fluxo do serviço de entrada vs curva de serviço [21]

Como é sabido, o tráfego UGS necessita de uma alocação de largura de banda constante. Com isto, os outros serviços necessitam de concorrerem pela restante largura de banda. Os resultados obtidos no gráfico da Figura 29 ilustram a largura de banda alocada para os

tipos de serviço rtPS, nrtPS e BE. Verifica-se que a arquitectura proposta é capaz de garantir a largura de banda existente para os tráfegos de maior prioridade bem como para os serviços de prioridade inferior.

3.4. “*DESIGN AND IMPLEMENTATION OF A QoS-INCLUDED WIMAX MODULE FOR NS-2 SIMULATOR*” DE AYMEN BELGHITH E LOUTFI NUAYMI

Este artigo [22] foi baseado na implementação criada pelo módulo da NIST para o ns-2. A contribuição feita pelos autores consiste na adição das classes de QoS bem como na gestão dos seus requisitos, mecanismos para oportunidades de pedidos *unicast* e de contenção, e ainda, algoritmos de escalonamento para as classes UGS, rtPS e BE.

3.4.1. GESTÃO DINÂMICA DE SERVIÇOS

O módulo implementado contém uma classe, denominada *ServiceFlow*, que identifica os requisitos dos serviços da ligação associada. Esta classe contém apenas o identificador de fluxo de serviço (SFID) e os parâmetros dos tipos de escalonamento dos fluxos de serviço (UGS, rtPS, ertPS, nrtPS ou BE). Não existe diferença entre as classes de QoS. Assim, os autores adicionaram os seguintes parâmetros:

- CID: representa o identificador da ligação;
- Prioridades de tráfego: define a prioridade atribuída a cada tipo de fluxo de serviço;
- Máxima taxa de transferência efectiva: define a informação do pico desta taxa do fluxo de serviço;
- Mínima taxa de transferência efectiva reservada: define a taxa mínima reservada do fluxo de serviço;
- *Jitter* tolerado: define a máxima variação do atraso de uma ligação;
- Atraso máximo: especifica o atraso máximo entre a recepção de um pacote e o envio deste mesmo pacote.

Depois dos parâmetros de fluxo de serviço estarem inicializados, a adição de um novo fluxo de serviço pode ser realizada. Para isto é necessária a troca de pacotes DSA

(*Dynamic Service Addition*). Estes pacotes são definidos pela norma 802.16, já estando definidos no módulo implementado pela NIST. Estes pacotes não contêm os parâmetros de fluxos de serviço, assim como o algoritmo de escalonamento não leva em consideração os requisitos das classes de QoS. Foram modificados estes pacotes DSA introduzindo-lhes algumas instruções:

- Quando um pedido DSA (DAS-REQ) é enviado, os parâmetros do fluxo de serviço é adicionado a este pacote como é definido na norma 802.16.
- Quando o pacote DSA-REQ é recebido, o fluxo de serviço e os parâmetros de QoS, da ligação de dados criada, são preenchidos pelos parâmetros recebidos pelo pacote DSA-REQ.
- Quando um pacote de resposta DSA (DAS-RSP) é enviado ou recebido, é efectuado o mesmo procedimento que o realizado nos pacotes DSA-REQ.

Estes diferentes passos estão ilustrados na Figura 30.

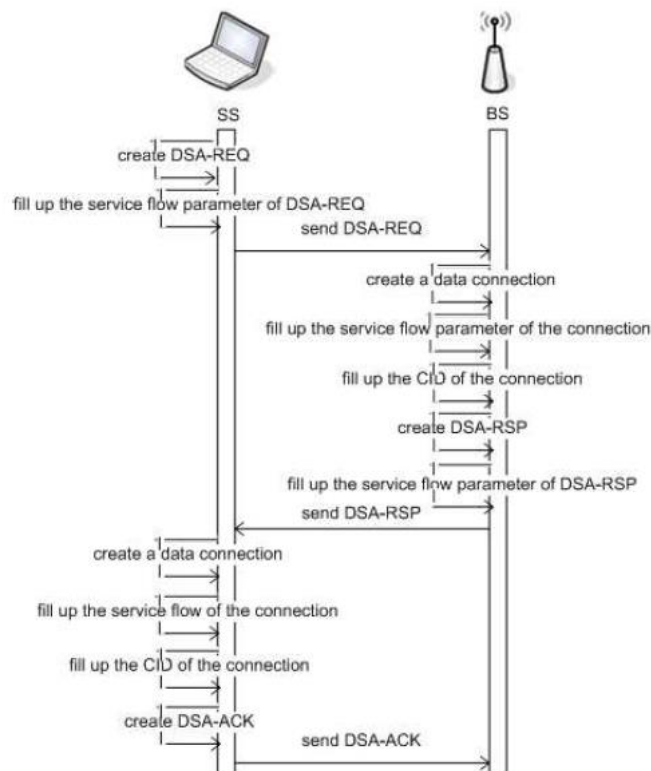


Figura 30 - Passos para a criação de um novo fluxo de serviço [22]

3.4.2. MECANISMOS PARA OPORTUNIDADE DE PEDIDOS

Oportunidade de Pedidos *Unicast*

Como é possível verificar na Figura 31, primeiro a BS determina uma lista de *polling unicast*. Uma SS que pertence a esta lista quando o frame corrente corresponde ao seu período de pedido de *polling unicast*. Depois, a BS aloca *bursts* de *uplink* com um código UIUC (*Uplink Interval Usage Code*). Quando uma SS recebe um *burst* de pedido de *polling unicast*, envia o pedido de largura de banda.

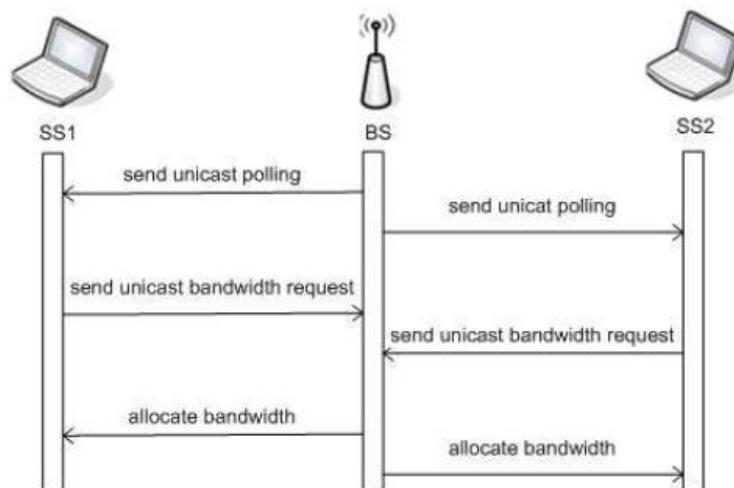


Figura 31 - Oportunidade de pedidos *unicast* [22]

Oportunidade de Pedidos de Contenção

Como é possível verificar na Figura 32, a BS aloca símbolos para oportunidades de pedidos de contenção no início de cada *subframe uplink*. Em todos os *frames*, todas as SSs, possuindo dados *uplink*, enviam pedidos de largura de banda por contenção. Foi modificada a implementação existente, de forma a proibir as SSs UGS e rtPS de usarem as oportunidades de pedidos por contenção. Além disso, estas SSs enviam os seus pedidos de largura de banda periodicamente e não em todos os *frames*.

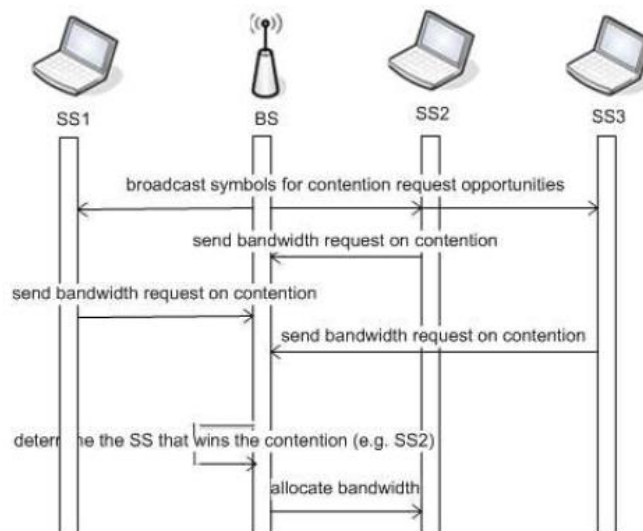


Figura 32 - Oportunidade de pedidos por contenção [22]

3.4.3. ESCALONAMENTO

A implementação da NIST não implementa uma diferenciação entre as classes de QoS. Neste caso a BS envia os pacotes no *subframe* de *downlink*. Depois, esta reserva todos os restantes símbolos para uma única estação usando um escalonamento do tipo *Round-Robin* (RR). Assim, em cada *frame*, apenas uma SS pode enviar os seus pacotes independentemente da classe de serviço. Na vista dos autores desta investigação, este algoritmo de escalonamento terá que ser alterado de forma a haver diferenciação dos fluxos de serviço, tendo em consideração os diferentes parâmetros de QoS. Para este efeito, foram consideradas três classes de QoS: UGS, rtPS e BE.

Para estas classes os autores propõem a combinação de alguns algoritmos já existentes para efectuar o escalonamento das classes de QoS. Como algoritmos utilizados para este efeito destacam-se o escalonador RR, que distribui equitativamente os recursos do canal a todas as SSs. O escalonador mSIR (*maximum Signal-to-Interference Ratio*), onde este aloca recursos de rádio às SSs que possuam os maiores níveis de SNR (*Signal-to- Noise Ratio*). O escalonador WRR (*Weighted round robin*) que é uma extensão do RR baseado em pesos estatísticos. O escalonador TRS (*Temporary Removal Scheduler*) bloqueia temporariamente SSs que possuam níveis de SNR menores que o limiar definido. Analisando estes métodos de escalonamento os autores propõem uma combinação entre os escalonadores TRS e os escalonadores RR e mSIR. Estas combinações são denominadas ao longo desta secção TRS+RR e TRS+mSIR [22][23].

Escalonamento da classe de QoS UGS

O escalonamento para a classe UGS está representado na Figura 33. A BS determina todas as SSs que possuam ligações UGS. A ligação que a BS irá começar a servir é escolhida aleatoriamente. Como cada SS é servida periodicamente, uma atribuição de um período para cada estação é recomendado (chamado período de escalonamento UGS). A partir deste período de escalonamento a BS determina se uma SS será servida no *frame* actual. Para cada ligação UGS que será servida, a BS determina o tempo de transmissão de acordo com o parâmetro de máxima taxa de transferência efectiva da ligação. Depois, a BS determina o número de símbolos a serem reservados para esta ligação UGS. Os símbolos disponíveis deverão ser suficientes para servir todas as ligações UGS. Após a reserva do número de símbolos estar determinada, a BS actualiza o mapa de *uplink* (UL-MAP) alocando um *burst* de *uplink*. O algoritmo de escalonamento UGS é realizado no início de cada *frame*.

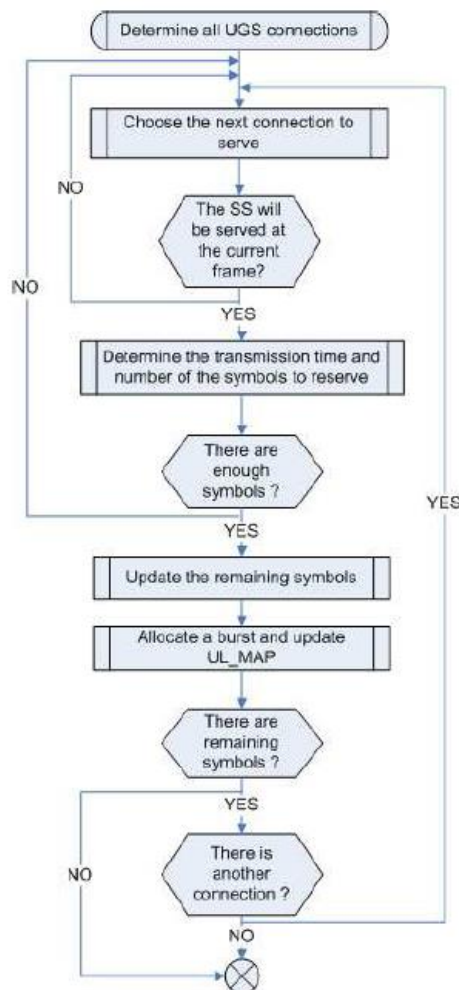


Figura 33 - Escalonamento UGS [22]

Escalonamento da classe de QoS rtPS

A proposta dos autores para o escalonamento da classe rtPS é ilustrada na Figura 34. A BS determina todas as SSs que possuam ligações rtPS. A escolha da próxima ligação a ser servida depende do algoritmo de escalonamento rtPS. Os algoritmos implementados para o escalonamento de classe rtPS são o RR, o mSIR, o WRR e o TRS. Se a SS não apresenta um pedido de largura de banda a satisfazer, a BS determina a próxima SS a servir. Caso contrário, a BS determina o tempo de transmissão e depois o número de símbolos a serem reservados para esta ligação rtPS. Se não existirem o número de símbolos suficientes, a BS aloca os restantes símbolos. Se a BS conseguir satisfazer inteiramente o pedido de largura de banda, esta remove o pedido. Caso este facto não se verifique, a BS actualiza a largura de banda para que seja servida no próximo *frame*. Depois da SS estar servida, a BS determina a próxima SS a ser atendida se ainda existirem símbolos excedentes.

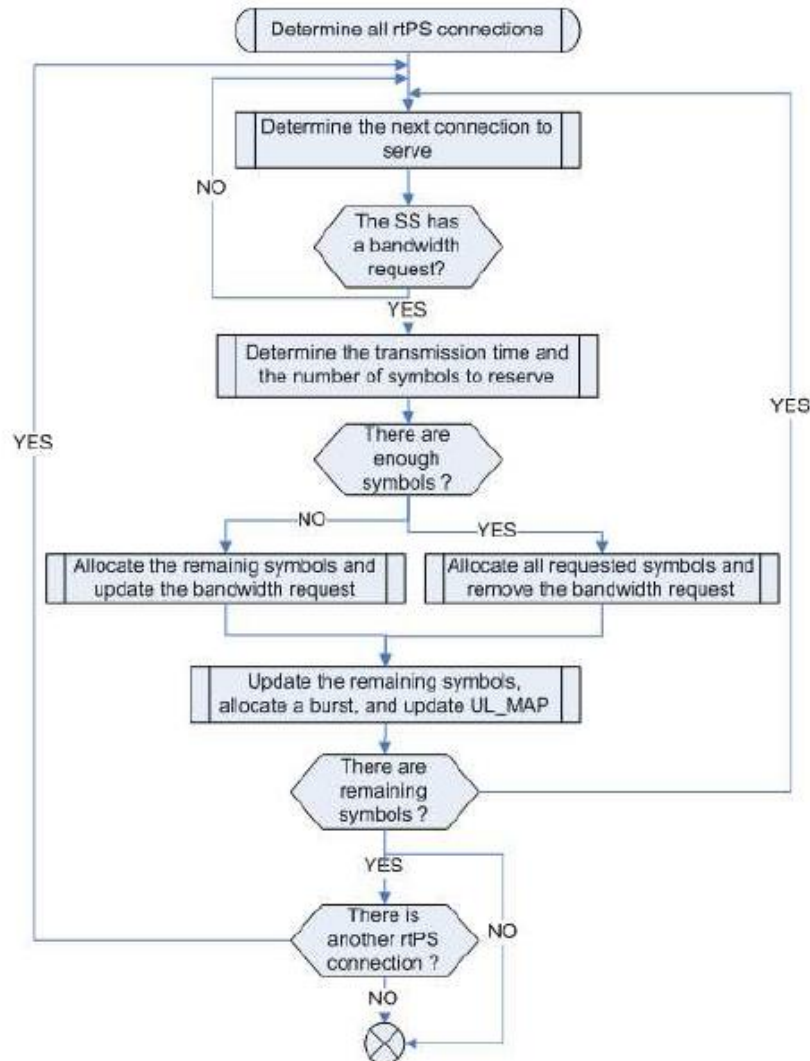


Figura 34 - Escalonamento rtPS [22]

Escalonamento da classe de QoS BE

A Figura 35 representa o funcionamento do escalonamento da classe BE. Primeiro, a BS determina todas as SSs que possuam ligações BE. A escolha da primeira ligação a ser servida é feita através do método RR. Se a SS não possuir um pedido de largura de banda, a BS verifica a próxima SS. Caso contrário, a BS determina o tempo de transmissão e o número de símbolos a reservar. Se não existirem símbolos suficientes, a BS aloca todos os símbolos restantes para a ligação que está a servir. Se a BS conseguir satisfazer o pedido de toda a largura de banda, remove este pedido. Se não for este o caso, a BS actualiza a largura de banda a ser alocada nos próximos *frames*. Depois de esta SS estar servida, a BS determina a próxima SS a ser servida, se ainda restarem símbolos.

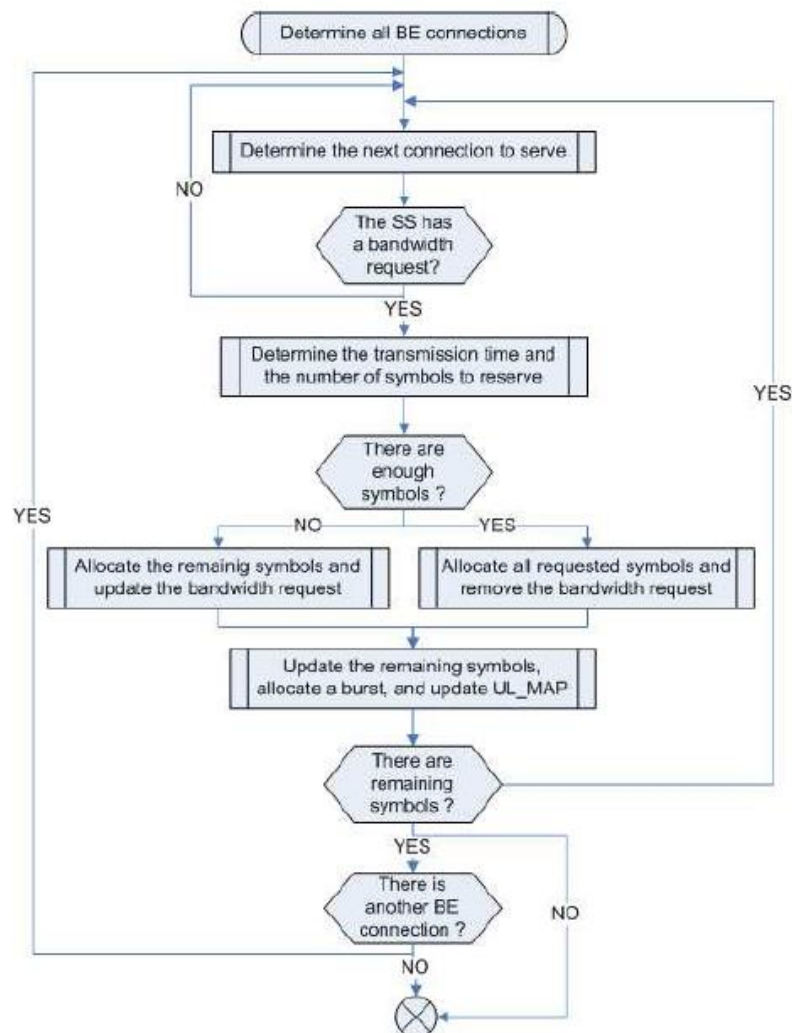


Figura 35 - Escalonamento BE [22]

3.4.4. RESULTADOS APRESENTADOS

Com o objectivo de avaliar a performance dos algoritmos de escalonamento apresentados, foram propostos três cenários de simulação, correspondendo aos três métodos de escalonamento sugeridos. Em cada cenário estão presentes na rede 5 SSs UGS, 9 SSs rtPS e 2 SSs BE.

Na primeira análise é estudado o comportamento do escalonador rtPS. É considerado o escalonador já existente no módulo da NIST (*Existing_RR*), bem como os novos escalonadores implementados, RR, mSIR, WRR, TRS+RR e TRS+mSIR. Neste cenário as cinco SSs UGS geram tráfego CBR (*Constant Bit Rate*) a uma taxa de 160kbit/s e as duas SSs BE geram tráfego FTP. A Figura 36 representa a taxa de transferência efectiva (*throughput*) das ligações rtPS em função do aumento da carga do tráfego rtPS submetida na rede.

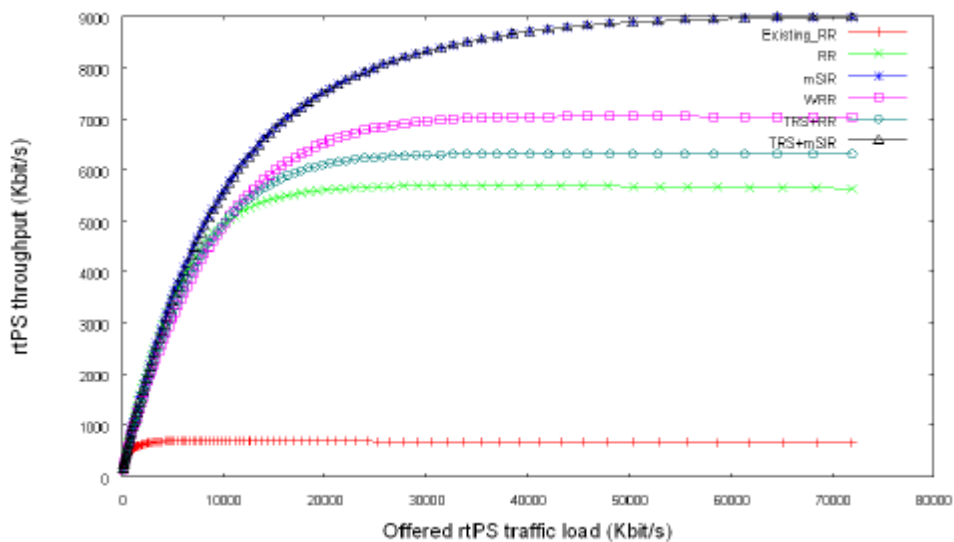


Figura 36 - Taxa de transferência efectiva rtPS vs Carga do tráfego rtPS oferecida [22]

A figura demonstra a baixa eficiência do escalonador *Existing_RR*. Este apresenta valores aproximadamente 9 vezes inferiores aos encontrados nos escalonadores implementados pelos autores. Observa-se que os escalonadores mSIR e TRS+mSIR superam os outros apresentando uma taxa de transferência de 9Mbit/s.

A Figura 37 apresenta o tempo de atraso encontrado nas ligações rtPS em função do aumento da carga do tráfego rtPS submetida na rede.

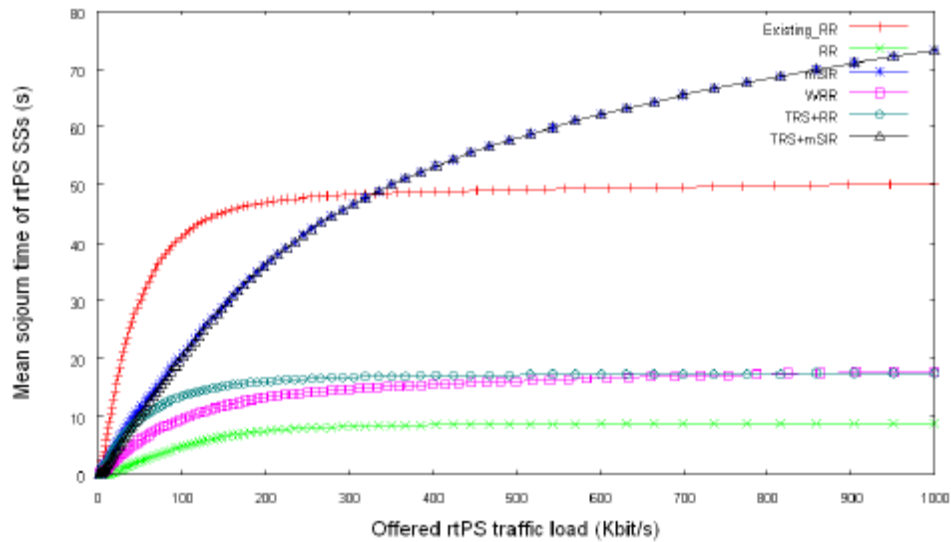


Figura 37 - Tempo de atraso (*mean sojourn time*) rtPS vs carga do tráfego rtPS oferecida

Verifica-se que o escalonador *Existing_RR* necessita de um alto nível de atraso médio para entregar um pacote de dados. Verifica-se também que os escalonadores mSIR e TSR+mSIR apresentam piores resultados que o encontrado no escalonador *Existing_RR*. Os escalonadores RR, WRR e TRS+RR são os que apresentam melhores resultados.

Numa segunda análise é estudado o comportamento do escalonador UGS. As 9 SSs rtPS geram tráfego com uma taxa de 80kbit/s, as duas SSs BE geram tráfego FTP. A Figura 38 ilustra a taxa de transferência das ligações UGS em função do aumento da carga do tráfego UGS submetida na rede.

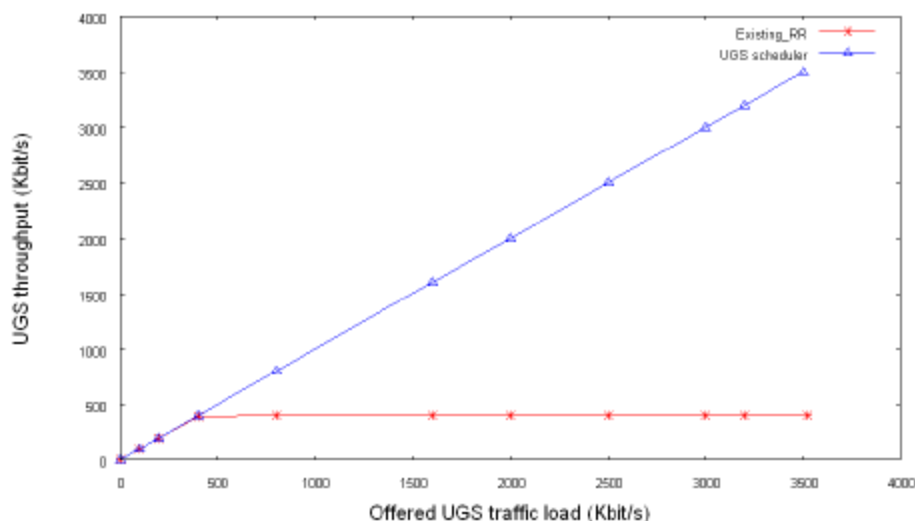


Figura 38 - Taxa de transferência efectiva UGS vs Carga do trafego UGS oferecida [22]

Verifica-se nesta análise que para cargas de tráfego baixas as taxas de transferência efectiva são iguais. O escalonador *Existing_RR* é incapaz de apresentar taxas de transferência efectiva superiores a 500kbit/s. O escalonador criado pelos autores comporta-se de acordo com os requisitos da norma, demonstrando que todas as SSs com tráfego UGS são inteiramente servidas.

Na Figura 39 é apresentado os atrasos encontrados para a segunda análise.

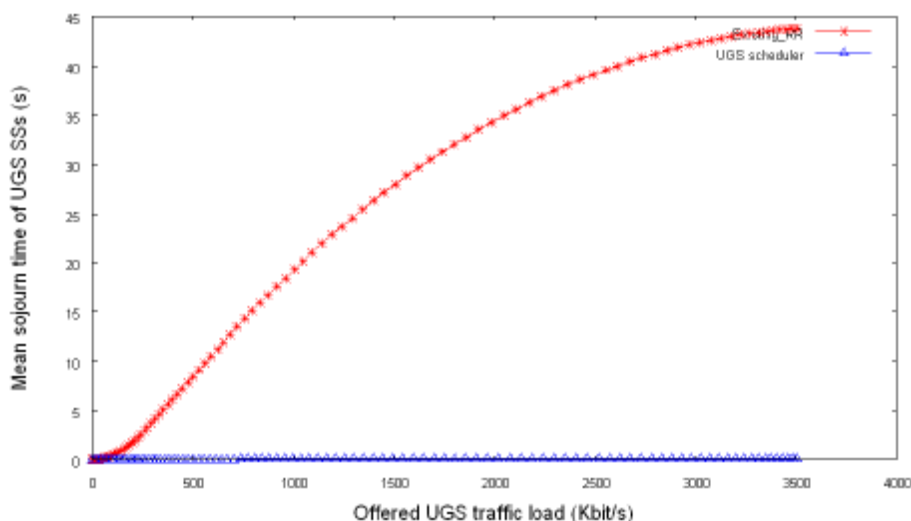


Figura 39 - Tempo de atraso (*mean sojourn time*) UGS vs carga do tráfego UGS oferecida

O escalonador criado pelos autores apresenta atrasos médios muito baixos, enquanto o escalonador já existente no módulo da NIST apresenta resultados insatisfatórios.

Na terceira análise realizada efectuou-se o estudo do comportamento do escalonador BE. A Figura 40 ilustra os resultados obtidos para um cenário composto por 9 SSs rtPS, com um tráfego de 80kbit/s e duas SSs BE com tráfego FTP. É apresentada a taxa de transferência efectiva em função do aumento da carga do tráfego UGS submetido na rede.

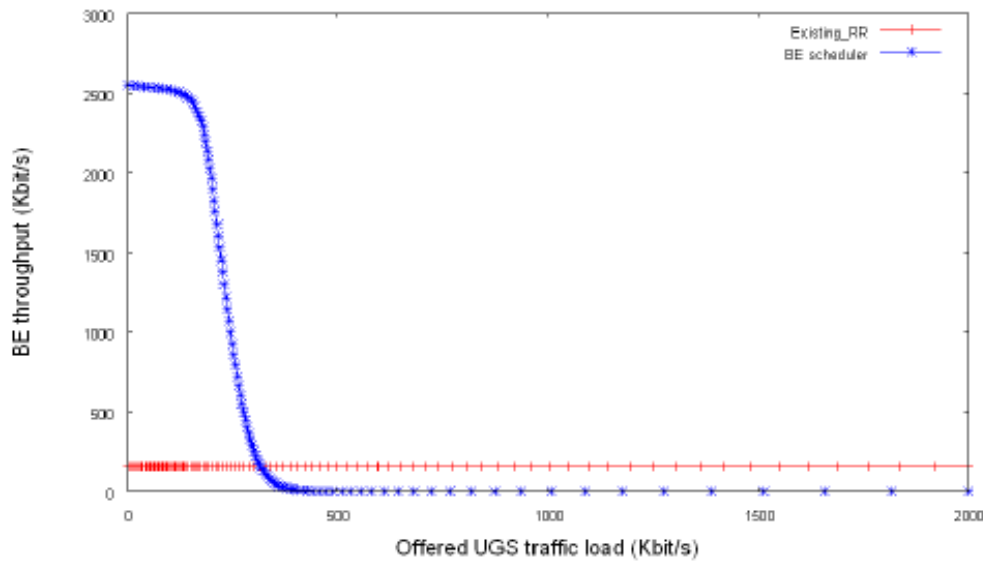


Figura 40 - Taxa de transferência efectiva UGS vs Carga do trafego UGS oferecida [22]

Neste gráfico analisa-se que o escalonador *Existing_RR* apresenta a taxa de transferência efectiva sempre constante, demonstrando que não é afectada pelo aumento da carga do tráfego UGS, significando um mau funcionamento para esta classe de pior prioridade. No caso do escalonador implementado neste artigo, verifica-se que, com o aumento da carga do tráfego UGS, a taxa de transferência efectiva diminui, demonstrando que a classe BE está a ser tratada com menor prioridade.

3.5. RESUMO

O terceiro capítulo tem como objectivo apresentar os trabalhos mais relevantes de investigação de QoS no seio da comunidade científica. Foram reunidos quatro trabalhos apresentado um resumo das suas principais características demonstrando também, as suas principais conclusões.

O primeiro trabalho “*Ensuring the QoS Requirements in 802.16 Scheduling*” propõe um mecanismo de escalonamento baseado conceptualmente numa aproximação por *Round-Robin* (RR) similar ao *Weighted Round-Robin* (WRR) em que cada ligação pode ser tratada por uma sessão separada.

O segundo trabalho “*Performance Analysis of the IEEE 802.16 Wireless Metropolitan Area Network*” propõe uma arquitectura que fornece suporte de QoS para o tráfego em tempo real com alta prioridade enquanto mantém o desempenho em níveis aceitáveis para

o tráfego de baixa prioridade. É proposto que na BS se acrescente uma parte onde é efectuado o escalonamento para a alocação da largura de banda (algoritmo de escalonamento que suporta todos os tipos de serviços transmitidos) e uma parte de controlo de admissão.

No terceiro trabalho “*An Integrated QoS Control Architecture for IEEE 802.16 Broadband Wireless Access Systems*” é proposta uma abordagem a dois níveis. Uma arquitectura de serviços integrados (IntServ) e uma arquitectura de serviços diferenciais (DiffServ). Vários artigos propõe algoritmos que se concentram nos problemas de QoS da camada MAC, mas pouco são aqueles que apresentam uma solução que garanta os requisitos de serviços de QoS para a camada superior. Neste artigo é proposta uma forma de garantir um controlo de *cross layer* QoS.

O quarto trabalho “*Design and Implementation of a QoS-included WIMAX Module for NS-2 Simulator*” é baseado num módulo de WIMAX já existente para simulação na ferramenta ns-2 levando a que a contribuição feita pelos autores consista na adição das classes de QoS bem como na gestão dos seus requisitos, mecanismos para oportunidades de pedidos *unicast* e de contenção, e ainda, algoritmos de escalonamento para as classes UGS, rtPS e BE. Para classe de QoS os autores propõem a combinação de alguns algoritmos já existentes para efectuar o seu escalonamento. Como algoritmos utilizados para este efeito destacam-se o escalonador RR (*Round-Robin*), o escalonador mSIR (*maximum Signal-to-Interference Ratio*), o escalonador WRR (*Weighted round robin*) e o escalonador TRS (*Temporary Removal Scheduler*).

4. FERRAMENTA DE SIMULAÇÃO – NS

A simulação é uma ferramenta essencial na pesquisa de redes de computadores, pois permite o desenvolvimento e a análise de novos protocolos e mecanismos em topologias complexas, sem a necessidade de implementá-las fisicamente. De todas as ferramentas disponíveis de forma gratuita para a simulação de redes de computadores, o *Network Simulator* (ns-2) é a ferramenta de maior popularidade entre os investigadores. Esta popularidade deve-se ao facto do ns-2 ser do domínio público e permitir a simulação de grande parte dos protocolos utilizados na Internet tanto em redes cabeadas como em redes sem fios.

4.1. PORQUÊ O NS?

Para que os resultados, apresentados neste documento, sejam fidedignos, foi efectuada uma pesquisa documental onde se investigou qual a ferramenta de simulação seria a mais adequada.

O ns-2 e o OPNET são as duas ferramentas mais populares e credíveis no que diz respeito à simulação de redes sem fios.

No ponto de vista dos investigadores o ns-2 fornece resultados muito similares quando comparado com a ferramenta OPNET. Sendo o ns-2 uma ferramenta *open source* e gratuita, este ganha uma maior relevância a nível académico, fazendo do ns-2 a ferramenta de simulação preferencial ao nível da investigação.

Do ponto de vista dos operadores de redes, por ser uma ferramenta que apresenta maiores funcionalidades, o conjunto completo da ferramenta OPNET pode ser o mais atractivo [24][25].

4.2. INTRODUÇÃO

Ns (versão 2) é um simulador orientado ao objecto dirigido à simulação de eventos discretos desenvolvido pela Universidade UC Berkely que simula uma grande variedade de protocolos de redes IP. É de grande utilidade na simulação de redes locais ou redes de grande área de cobertura. Por haver poucos manuais que facilitam a utilização, esta ferramenta apresenta uma grande dificuldade para utilizadores que estejam a usar este simulador pela primeira vez.

O ns é do domínio público apresentando diversas funcionalidade bem como um código *open source* capaz de ser alterado. Algumas das suas funcionalidades são [26]:

- Elementos para topologias de redes – nós e ligações ponto a ponto.
- *Routing* – *unicast*, *multicast* e hierárquico.
- Esquemas de filas – *drop-tail*, RED, CBQ, FQ e SFQ.
- Protocolos de transporte – UDP e TCP.
- Geradores de tráfego – CBR, exponencial e *Pareto*.
- Aplicações – *Web cash*.
- Mobilidade básica – redes ad hoc, IP móvel.

4.3. FUNCIONAMENTO GERAL

O ns é um simulador orientado ao objecto, escrito em linguagem C++, com um interpretador OTcl (*object-oriented Tcl*) como *frontend*. O simulador suporta uma hierarquia de classes em C++ e uma hierarquia de classes similar dentro do interpretador OTcl [27]. Estas duas hierarquias estão relacionadas uma com a outra como ilustra a Figura 41.

O ns usa estas duas linguagens porque o simulador necessita de efectuar duas tarefas diferentes. Por um lado, a simulação de protocolos muito detalhados requer um sistema de programação que consiga manipular, de uma forma eficiente, bytes, cabeçalhos de pacotes e implementar algoritmos que corram sobre uma grande quantidade de dados. Por outro lado, uma grande parte de investigações de rede envolve uma ligeira variação de parâmetros ou configurações, ou ainda a exploração de múltiplos cenários. Neste caso, o tempo de interacção (mudança de modelo ou reinício da simulação) é o mais importante.

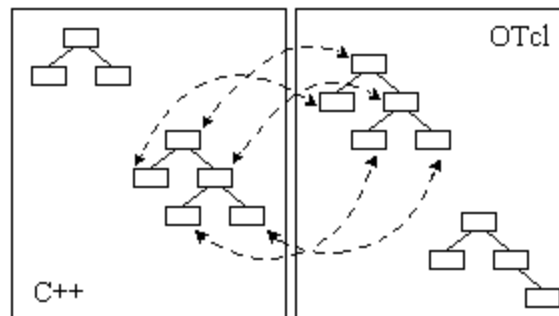


Figura 41 - C++ e OTcl [27]

Devido a estas diferenças o ns necessita de ambas as linguagens, onde o C++ é mais rápido no funcionamento mas mais lento no que diz respeito a configurações e alterações dos cenários, fazendo desta linguagem ajustada à implementação de protocolos detalhados. O OTcl é executado de um modo mais lento por ser interpretado, mas apresenta resultados muito interessantes no que toca a interactividade tornando esta linguagem prática para configurações das simulações [28].

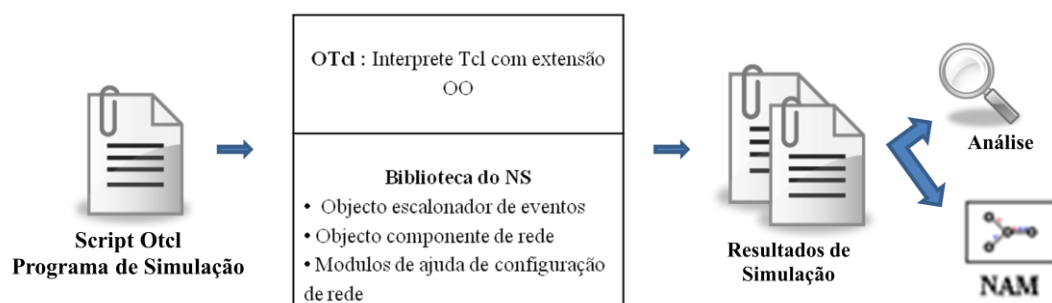


Figura 42 - Vista simplificada do ns-2 do ponto de vista do utilizador

Como mostra a Figura 42, o ns é um interpretador de *script* em Tcl orientado ao objecto (OTcl) tendo também um gestor de eventos, bibliotecas de componentes de rede e ainda bibliotecas de modelos de instalação das ligações de redes. Por outras palavras, para utilizar o ns é necessário programar na linguagem de *script* OTcl. Para instalar e correr uma simulação, o utilizador deverá escrever um *script* OTcl e iniciar um gestor de eventos, definir a topologia de rede usando os objectos de rede e funções das ligações de redes presentes nas bibliotecas. Para além disso é necessário indicar ainda o início e o término da transmissão de pacotes nas fontes de tráfego através do gestor de eventos.

Apesar desta programação em OTcl ser complexa para quem se está a iniciar neste tipo de linguagem, depois do utilizador ficar familiarizado com a programação destas *scripts* irá compreender que estes modelos incluídos nas bibliotecas acabam por facilitar a tarefa de simular os mais diversos tipo de redes, tanto em redes com cabos com em redes *wireless*.

Um dos aspectos mais importantes do ns, para além dos objectos de rede é o gestor de eventos. Um evento em ns é composto por um identificador de pacote (*packet ID*), que é único para cada pacote com um certo tempo de escalonamento, e um apontador para um objecto que controla esse mesmo evento. No ns, o gestor de eventos acompanha sempre o tempo de simulação lançando os eventos numa fila, invocando os componentes da rede apropriados, e levando estes a realizarem a acção necessária com o apontador associando a este evento.

A Figura 43 ilustra a arquitectura geral do ns-2.

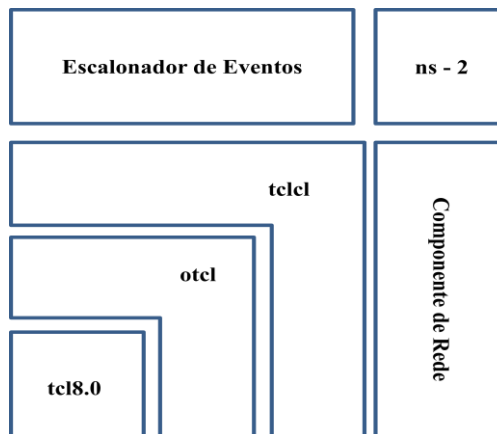


Figura 43 - Vista da arquitectura geral do NS

A criação e a execução de simulações pelo utilizador são feitas em tcl usando os objectos de simulação presentes nas bibliotecas Otcl. O escalonador de eventos e a maioria dos componentes de rede são implementados em C++ estando disponíveis para o Otcl através de uma ligação Otcl implementada pelo tclcl. Todo este conjunto forma a ferramenta ns-2.

4.4. PROCESSO DE SIMULAÇÃO

Como já foi referido o ns-2 é assente na simulação de eventos discretos. A sua topologia de simulação, de uma forma genérica, consiste em nós de rede interligados por ligações com uma certa largura de banda, atraso e uma estratégia de filas. Nas secções seguintes são apresentados os aspectos mais importantes na criação de uma topologia de rede para simulação com o ns-2.

4.4.1. ESCALONADOR DE EVENTOS

O funcionamento do escalonador de eventos pode ser descrito da seguinte forma: selecciona o próximo evento a executar, executa-o até à sua conclusão e por fim retorna da execução do evento. Actualmente apenas um evento pode ser executado a um certo instante. Se mais de um evento estiver escalonado para ser executado ao mesmo tempo, a selecção será efectuada pela forma “primeiro escalonado – primeiro expedido” [27].

Os principais utilizadores do escalonador de eventos são componentes de rede que simulam atrasos no manuseamento de pacotes ou que necessitam de temporizadores. A Figura 44 mostra cada objecto de rede usando um escalonador de eventos.

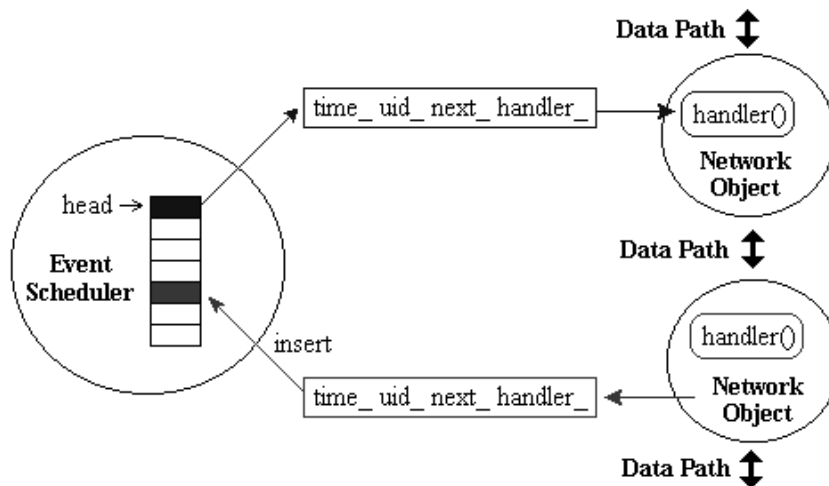


Figura 44 - Escalonador de eventos discretos [27]

Note-se que um objecto de rede que emite um evento, inserindo-o na falta do escalonador, é o que trata o mesmo evento mais tarde no tempo escalonado. Também é de realçar que o curso (*path*) de dados entre objectos de rede é diferente do curso do evento.

O ns utiliza dois tipos de escalonador de eventos, escalonador em tempo real e outro em tempo não real. No escalonador em tempo não real estão disponíveis três implementações (lista, pilha e agenda). A implementação em agenda é a que está definida por omissão no simulador. O escalonador em tempo real serve para emulações, permitindo ao simulador interagir com uma rede real.

Outra utilidade do escalonador de eventos é a selecção e activação de eventos de simulação, associados a aplicações, tais como o início e o término de um FTP. Um objecto de escalonamento de eventos tem também funções de escalonamento de simulação como por exemplo “*at time <string>*” que resulta num evento especial chamado *AtEvent* que ocorre num tempo de simulação específico.

4.4.2. NÓS E ROUTING

Um nó é composto por um objecto de entrada de nó e classificadores como mostra a Figura 45. Existem dois tipos de nós no ns. Um nó *unicast* que possui um classificador de endereços que faz *unicast routing* juntamente com um classificador de porta. E um nó *multicast* (Figura 46) que possui um classificador de pacotes *multicast* e um classificador de *routing multicast*.

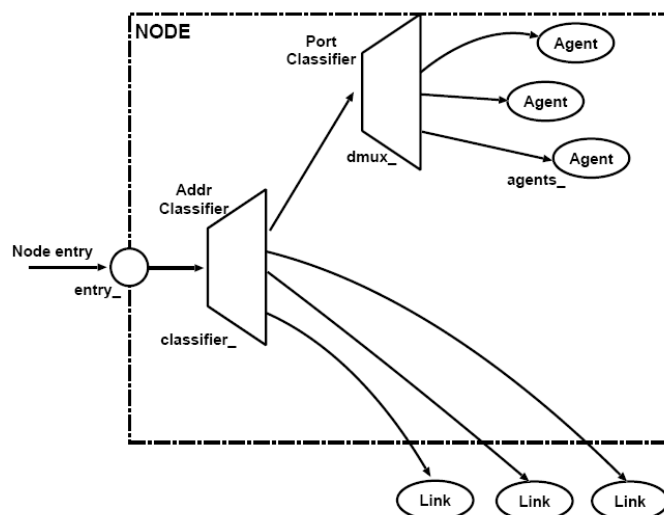


Figura 45 - Estrutura interna de um nó *unicast* [28]

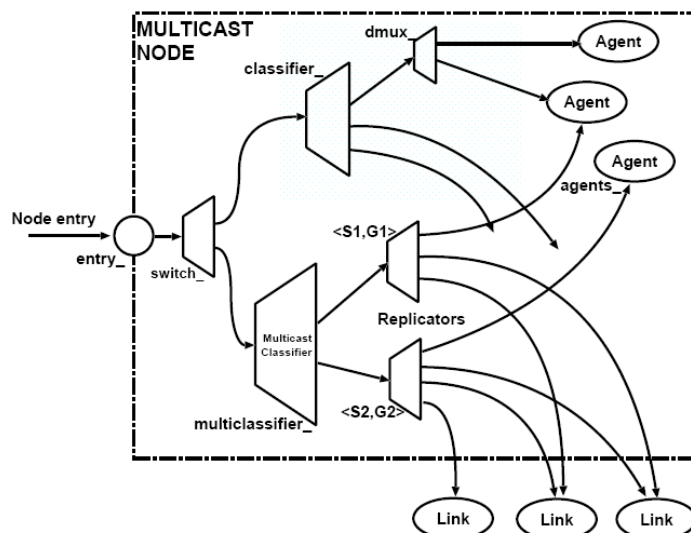


Figura 46 - Estrutura interna de um nó *multicast* [28]

No ns, os nós são, por omissão, do tipo *unicast*. Para criar nós *multicast* é necessário modificar os atributos do nó na *script* TCL. Para haver troca de pacotes entre os nós é necessário que agentes estejam ligados a cada nó. Um agente representa o ponto final onde os pacotes de rede são construídos ou consumidos, sendo usados na implementação de protocolos em diferentes camadas. Estes agentes podem ser agentes de *routing* ou agentes de envio/recepção para pacote TCP ou UDP.

4.4.3. FILAS

As filas representam locais onde os pacotes podem ser retidos (ou desprezados). O escalonamento dos pacotes refere-se ao processo de decisão utilizado para escolher quais os pacotes que devem ser seguros ou desprezados. O *Buffer* de gestão refere-se a qualquer disciplina específica usada para regular a ocupação de uma determinada fila. Actualmente estão disponíveis, no ns-2, filas do tipo *drop-tail*, RED (*Random Early Detection*) *Queue*, CBQ (*Class Based Queueing*), e variantes de *Fair Queueing* incluindo o FQ (*Fair Queueing*), *Stochastic Fair Queueing* (SFQ), e o *Deficit Round-Robin* (DRR) [28].

Cada tipo de fila é representado por um objecto de fila. Este objecto é uma classe geral capaz de reter, marcar ou descartar pacotes, enquanto estes se movimentam ao longo da topologia de rede. Cada objecto de fila é descrito da seguinte forma:

- Objecto *drop-Tail*: os objectos *drop-tail* são uma subclasse do objecto de fila que implementa um simples método de fila, FIFO (*First in First Out*);
- Objecto FQ: estes objectos representam uma subclasse do objecto de fila que implementa o método *Fair queueing*;
- Objecto SFQ: os objectos SFQ representam uma subclasse do objecto de fila que implementa um método *Stochastic Fair Queueing*;
- Objecto DRR: são uma subclasse do objecto de fila que apresentam um escalonamento *deficit round robin*. OS objectos DRR implementam este escalonamento entre os diversos fluxos. Ao contrário de outros objectos multi-fila, este objecto de fila implementa um único espaço de buffer partilhado para os seus diferentes tipos de fluxos;
- Objecto RED: os objectos RED são uma subclasse do objecto de fila que implementam uma aleatória detecção precoce de *gateways*. O objecto pode ser configurado para descartar ou marcar pacotes;
- Objecto CBQ: são uma subclasse de objectos de fila que implementam um método de fila baseado em classes.

4.4.4. LIGAÇÕES

Tal como os nós, as ligações são objectos compostos de grande importância no simulador ns. Quando é criada uma ligação simples entre dois nós são indicadas características como

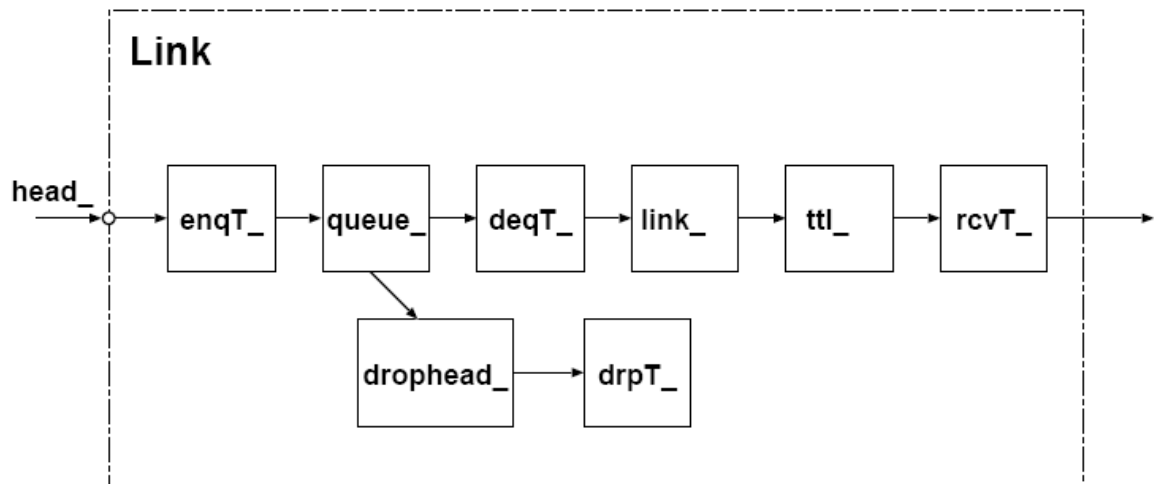


Figura 47 - Composição de uma ligação unidireccional [28]

a largura de banda, o atraso desta ligação e ainda o tipo de fila. A Figura 47 mostra a estrutura interna de uma ligação. Ao criar a ligação cinco variáveis de instância são concebidas :

- `head_` : ponto de entrada da ligação. Aponta para o primeiro objecto na ligação.
- `queue_` : reporta-se ao principal elemento de fila da ligação. Usualmente, cada ligação apresenta uma fila [28].
- `link_` : reporta-se ao elemento que actualmente modula a ligação em termos de características com atrasos ou largura de banda da ligação.
- `ttl_` : reporta-se ao elemento que manipula o tempo de vida (*time to live* - TTL) de cada pacote
- `drophead_` : reporta-se a elementos que processam perdas na ligação.

Em adição, se for usado o comando em TCL `$traceALLFile`, o procedimento irá adicionar elementos de análise de pacotes na ligação, tais como:

- `enqT_` : reporta-se ao elemento que analisa pacotes que entram na fila.

- `deqT_` : reporta-se ao elemento que analisa pacotes que saem da fila.
- `drpT_` : reporta-se ao elemento que analisa pacotes que se perdem na fila.
- `rcvT_` : reporta-se ao elemento que analisa pacotes recebidos no próximo nó.

4.4.5. PACOTE NS

Um pacote em ns é composto por uma pilha de cabeçalhos e um espaço de dados opcional (Figura 48). Um formato do cabeçalho de pacote é iniciado quando um objecto de simulação é criado, onde é definida a pilha de todos os cabeçalhos registados, tais como cabeçalhos comum (`cmn`), IP, TCP, RTP (UDP usa cabeçalhos RTP) e *trace*, sendo o cabeçalho de cada pilha gravado. Isto significa que independentemente de um cabeçalho ser usado, a pilha composta por todos os cabeçalhos é criada quando um pacote é alocado por um agente, possibilitando que qualquer objecto de rede possa aceder a qualquer cabeçalho [27].

Normalmente, um pacote apenas contém a pilha de cabeçalhos (e um apontador para um espaço de dados que é nulo). É possível que um pacote possa transportar dados (a partir de uma aplicação), atribuindo um espaço de dados, mas muito poucas aplicações ou implementações apoiam este processo.

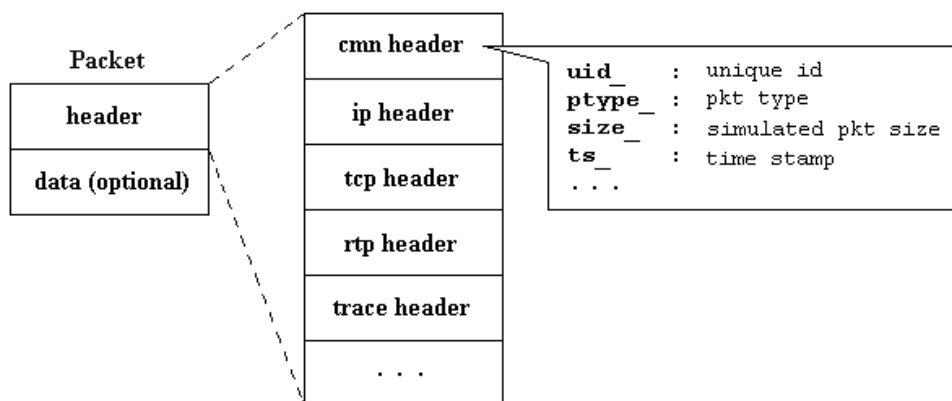


Figura 48 - Formato de um pacote no ns [27]

4.5. ROUTING

No ns-2, para modelos de rede *wireless*, existem quatro tipos de protocolos de routing ad-hoc [28]:

- DSDV (*Destination Sequence Distance Vector*): Neste protocolo de *routing* existe troca de mensagens entre nós vizinhos. Um pacote onde a rota para o seu destino não é conhecida, permanece em cache enquanto são enviadas inquirições de *routing*. Os pacotes são armazenados até as respostas de routing do destino surgirem. Existe um tamanho máximo de espaço no buffer para o armazenamento dos pacotes que estão à espera de respostas
- DSR (*Dynamic Source Routing*): Este protocolo possui um agente que verifica os pacotes de dados sobre informações de rotas de origem. Este encaminha o pacote com as informações encontradas. No caso de não existirem informações de *routing*, este protocolo, armazena o pacote e envia inquirições sobre a rota.
- TORA (*Temporally-Ordered Routing Algorithm*): Neste protocolo, em cada nó é executado uma cópia do algoritmo TORA para cada destino. Quando um nó necessita de uma rota para um determinado destino, envia por *broadcast* uma mensagem de inquirição contendo o endereço de destino da rota.
- AODV (*Adhoc On-demand Distance Vector*): É a combinação dos protocolos de *routing* DSDV e DSR. O AODV possui os mecanismos de descoberta de rota e manutenção do protocolo DSR, utilizando o routing salto-a-salto, sequencias de números e demarcações do DSDV.

4.6. ANÁLISE DOS RESULTADOS

Quando a simulação acaba, o NS tem a capacidade de produzir ficheiros *output* com informação detalhada dos dados da simulação. Para que tal aconteça é necessário especificar no *script* Tcl as instruções necessárias para que a informação seja registada. Estes dados podem ser usados para análises da simulação ou como input para uma ferramenta gráfica chamada *Network Animator* (NAM).

4.6.1. *TRACE FILE*

O ns disponibiliza um ficheiro de *output* que contém toda a informação necessária para se realizar uma análise da simulação efectuada (Figura 49).

Este ficheiro tem um formato universal para que os investigadores consigam desenvolver scripts em diversas linguagens de programação (*awk*, *perl*, etc.) fazendo assim a análise desejada (Tabela 6).

Tabela 6 - Explicação do *trace file* [29]

r	0.514	0	1	cbr	500	----	0	0.0	0.1	0	1
Event	Event time	Link btw node 0 & 1	Pkt type	Size byte s	flag	Flo w id	Src addr of pkt	Dst addr	Seq no	Pkt id	

```
+ 0.5 0 1 cbr 500 ----- 0 0.0 2.0 0 0
- 0.5 0 1 cbr 500 ----- 0 0.0 2.0 0 0
+ 0.505 0 1 cbr 500 ----- 0 0.0 2.0 1 1
- 0.508 0 1 cbr 500 ----- 0 0.0 2.0 1 1
+ 0.51 0 1 cbr 500 ----- 0 0.0 2.0 2 2
+ 0.515 0 1 cbr 500 ----- 0 0.0 2.0 3 3
- 0.516 0 1 cbr 500 ----- 0 0.0 2.0 2 2
r 0.518 0 1 cbr 500 ----- 0 0.0 2.0 0 0
+ 0.518 1 2 cbr 500 ----- 0 0.0 2.0 0 0
- 0.518 1 2 cbr 500 ----- 0 0.0 2.0 0 0
+ 0.52 0 1 cbr 500 ----- 0 0.0 2.0 4 4
```

Figura 49 Ex de um *trace file* [29]

Recentemente surgiu um novo formato do *trace file* para redes *wireless*. Este formato foi o utilizado nas simulações realizadas no decorrer deste trabalho. Este novo formato é composto por informação bastante mais detalhada que o formato original.

```
s -t 163.001503520 -Hs 0 -Hd -2 -Ni 0 -Nx 300.00
-Ny 500.00 -Nz 0.00 -Ne -1.000000 -Nl AGT -Nw ---
-Ma 0 -Md 0 -Ms 0 -Mt 0 -Is 0.0 -Id 2.0 -It cbr -
Il 200 -If 1 -Ii 77 -Iv 32 -Pn cbr -Pi 32 -Pf 0 -
Po 0
```

Este extrato refere-se apenas a uma linha do *trace file*. Contem informação relativa ao tipo de evento (*s* (*send* – envio), *r* (*receive* – recepção), *d* (*drop*), *f* (*forward*)), ao tempo em que ocorreu o evento, aos nós de origem e destino, ao tipos de pacotes tanto no nível IP, MAC e no nível de aplicação.

4.6.2. NETWORK ANIMATOR – NAM

O NAM é uma ferramenta baseada em Tcl/Tk para a observação animada de simulações de rede. Suporta diferentes topologias, animações ao nível dos pacotes e ainda várias ferramentas de inspecção de dados.

Para uma boa observação de uma simulação que se pretenda testar, o NAM é um poderoso auxílio proporcionando uma percepção da topologia e actividade na rede. Com o NAM é possível observar os pacotes a partirem de um determinado nó chegando ao destino podendo-se fazer uma monitorização dos caminhos dos pacotes e da ligação entre vários nós. A Figura 50 ilustra a ferramenta NAM, onde se observa o decorrer de uma simulação com três nós ligados entre si. Cada nó possui uma aplicação CBR sobre o protocolo TCP.

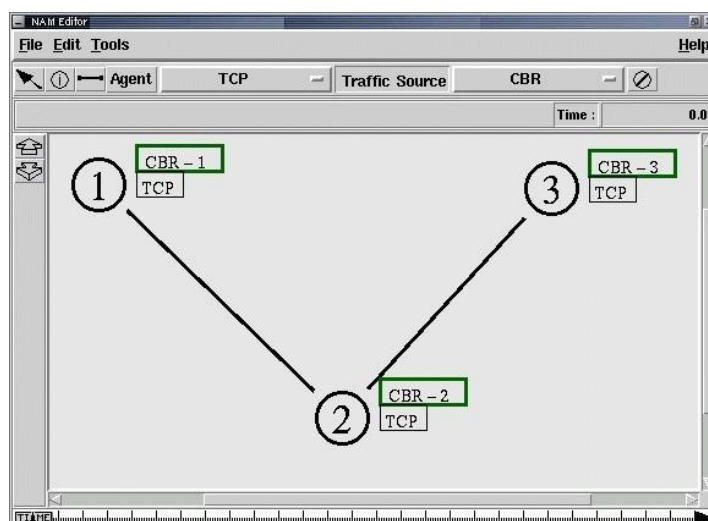


Figura 50 - NAM – *network animator*

4.7. AWK

De forma a ser possível interpretar os *trace files* foi utilizada a linguagem de *scripting* AWK. O AWK é uma linguagem de programação de uso geral sendo projectado para o processamento de dados baseados em texto, podendo ser utilizado na interpretação de ficheiros ou *streams* de dados. Esta linguagem é de extrema utilidade quando se pretende manipular ficheiros com formatos universais, possuindo um tipo de linguagem similar ao C++.

A estrutura de uma *script* AWK é bastante simples composta tipicamente por três partes. O início da *script*, onde são introduzidas as variáveis globais a ser utilizadas, o corpo da *script*, que será executado para cada linha do *trace file*, e por fim, o fecho da *script*, onde tipicamente serão impressos os resultados da *script*.

```
BEGIN {inicialização das variáveis}
{
  O conteúdo será executado a cada linha do trace
  file;
  $n -> retorna a variável presente na coluna n do
  trace file.
}
END{Imprime os dados}
```

No âmbito deste trabalho foi necessário criar *scripts* awk para retirar dos *trace files*, das simulações efectuadas, valores como a taxa de transferência efectiva e o atraso médio (secção 6.1). Os extractos de código seguintes procuram demonstrar, de uma forma resumida, como este processo é realizado.

Depois da inicialização de variáveis globais, no início da *script*, o corpo da mesma inicia-se com a atribuição de variáveis às colunas do *trace file* necessárias ao cálculo da taxa de transferência efectiva e do atraso médio.

```
if ($2 == "-t") {
    event = $1
    time = $3
    node_id = $5
    flow_id = $39
    pkt_id = $41
    pkt_size = $37
    flow_t = $45
    level = $19
}
```

De seguida é possível armazenar os dados referentes aos tempos de envio de um pacote.

```
# Guarda o tempo de envio dos pacotes
if (level == "AGT" && flow_id == flow && fonte
== src && event == "s") {
    (...)
    sendTime[pkt_id] = time
    (...)
}
```

Neste ponto está criado um vector que vai armazenar todos os tempos de envio de pacotes no nível de aplicação com o identificador de fluxo igual ao que se pretende calcular e com o endereço de origem que se pretende obter.

Da mesma forma que se armazenam os tempos de envio dos pacotes, é possível armazenar os tempos de chegada dos pacotes bem como a quantidade de bits que é recebida no destino.

```
# Actualiza o tamanho total de pacotes
recebidos e o seu tempo de chegada
if (level == "AGT" && flow_id == flow && node_id
== dst && event == "r" && fonte == src) {
    (...)
    # armazena o tamanho do pacote recebido
    recvdSize += pkt_size
    (...)
    # armazena o tempo de recepção do pacote
    recvTime[pkt_id] = time
    #printf("pacote recebido: %d no tempo
    (...)
```

No fecho da *script*, já é possível imprimir os valores de atrasos e das taxas de transferências.

```
END {
    # Cálculo do atraso médio
    delay = avg_delay = recvdNum = 0
    (...)
    delay += recvTime[i] - sendTime[i]
    recvdNum ++
}
(...)
if (recvdNum != 0) {
    avg_delay = delay / recvdNum
}

(...)
# Cálculo da taxa de transferência efectiva
Tput = (recvdSize/(tempoFim-
tempoInicio))*(8/1000))
```

4.8. CONFIGURAÇÃO DE CENÁRIOS DE SIMULAÇÃO

Para a descrição do funcionamento de uma simulação com o ns-2 foi usado o exemplo de uma configuração redes simples [27], cujo cenário é ilustrado na Figura 51. Esta rede é composta por quatro nós (n0, n1, n2, n3). As ligações duplas entre n0 e n2, e entre n1 e n2 possuem 2 Mbps de largura de banda e 10ms de atraso. A ligação dupla entre n2 e n3 possui 1.7 Mbps de largura de banda e um atraso de 20ms. Cada nó usa uma fila do tipo *dropTail* em que o tamanho máximo é 10. Um agente TCP está anexado ao n0 e uma ligação é estabelecida até a um agente TCP *Sink* que está anexado ao n3. Este agente TCP

Sink é utilizado como receptor numa ligação TCP sendo capaz de enviar mensagens de confirmação de recepção dos pacotes. Por omissão, o tamanho máximo de um pacote que um agente TCP consegue gerar é 1KByte. Um agente TCP *Sink* gera e envia pacote ACK ao emissor (agente TCP) e liberta o pacote recebido. Um agente UDP está anexado ao nó n1 que está ligado a um agente NULL anexo ao nó n3. Um agente NULL, ao contrário do agente *Sink*, apenas liberta o pacote recebido. Geradores de tráfegos FTP e CBR estão anexados a agentes TCP e UDP, respectivamente. O CBR é configurado para gerar pacotes de 1KByte à taxa de 1Mbps, sendo escalonado para iniciar a 0.1s e finalizar a 4.5s. O FTP é escalonado para iniciar a 1s e finalizar ao 4º segundo.

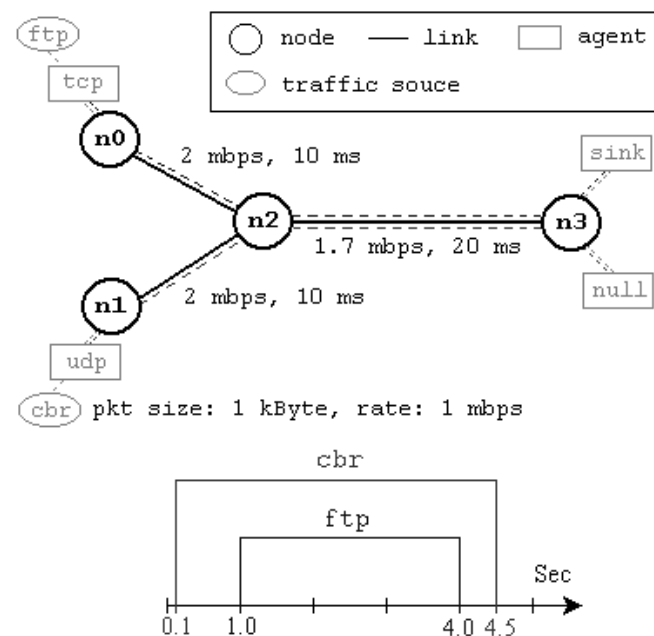


Figura 51 - Simples topologia de rede e o cenário de simulação [27]

Para executar a simulação o ns-2 requer uma script Tcl que deverá ter algumas das seguintes construções [30][31][32]:

O início de qualquer simulação requer que seja criado um objecto de simulação do ns.

```
set ns [new Simulator]
```

É a instrução usada para gerar uma instância no simulador ns. Esta linha de código efectua as seguintes funções:

- Inicia o formato do pacote;

- Cria um escalonador, que por omissão é o escalonador de agenda;
- Selecciona o formato de endereçamento por omissão.

Este objecto de simulação possui funções membro que permitem executar os seguintes procedimentos:

- Criação de objectos como nós e ligações;
- Criação de objectos de ligação de rede (agentes);
- Configuração de parâmetros de componentes de redes;
- etc.

Para permitir a análise dos eventos associados à troca de mensagens e possibilitar a visualização animada da simulação realizada, são usados ficheiros de logs.

```
set f [open out.tr w]
$ns trace-all $f
set nf [open out.nam w]
$ns namtrace-all $nf
```

Para a análise de simulações em ambientes wireless foi introduzido um novo formato do *trace file* onde cada linha tem informações bastante mais detalhadas. Para que este novo formato seja empregue ao *trace file* é necessário utilizar o seguinte comando no código TCL:

```
$ns use-newtrace
```

Estas linhas indicam ao simulador para gravar todas trocas de mensagens existentes no processo de simulação guardando nos formatos *output Trace* e NAM. São abertos os ficheiros out.tr e out.nam em modo de escrita sendo apontados pelas variáveis *f* e *nf* respectivamente.

Na função abaixo representada são também especificados processos a serem executados depois da simulação ter terminado. Neste caso a aplicação NAM será executada com os dados guardados no ficheiro out.nam.

```
proc finish {} {
    global ns f nf
    $ns flush-trace
    close $f
    close $nf
}
```

```

    puts "running nam..."
#execução do programa NAM
    exec nam out.nam &
    exit 0
}

```

A função *proc finish* é chamada após a simulação terminar pelo comando:

```
$ns at 5.0 "finish"
```

A criação dos nós efectuada pelas linhas de código seguintes sendo cada nó associado a uma variável (n0, n1, etc.).

```

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]

```

Cada uma das linhas de código descritas, de seguida, criam duas ligações simples entre os respectivos nós atribuindo-lhes as características especificadas de largura de banda e de atraso. O tipo de fila usado de output de um nó é implementado como parte de uma ligação. A fila usada nestes casos foi a *dropTail*.

```

$ns duplex-link $n0 $n2 2Mb 10ms DropTail
$ns duplex-link $n1 $n2 2Mb 2ms DropTail
$ns duplex-link $n2 $n3 1.7Mb 20ms DropTail

```

Este extracto de código é utilizado para dispor as ligações no programa de visualização NAM. A última linha efectua a monitorização da fila para a ligação n2-n3.

```

$ns duplex-link-op $n0 $n2 orient right-up
$ns duplex-link-op $n1 $n2 orient right-down
$ns duplex-link-op $n2 $n3 orient right

#Monitorar a fila para ligação n2-n3 (no NAM)
$ns duplex-link-op $n2 $n3 queuePos 0.5

```

A criação de uma ligação TCP requer que um nó tenha um agente do tipo fonte que inicia a aplicação e outro do tipo destino (*Sink*) que atende ligação.

Com o intuito de configurar a ligação entre os nós n0 e n3 utiliza-se o seguinte extracto de código a ligação TCP fica configurada. Depois de criado um agente TCP este é anexado ao n0. O mesmo processo é efectuado para o n3 com o agente TCPSink. Após os agentes estarem anexados aos nós basta conectar os dois agentes criados. A instrução *set fid_ 1* serve para análises feitas pós simulação onde atribui ao tráfego TCP o número 1.

```

# configuração da ligação TCP
set tcp [new Agent/TCP]
$tcp set class_ 2

```

```

set sink [new Agent/TCPSink]
$ns attach-agent $n0 $tcp
$ns attach-agent $n3 $sink
$ns connect $tcp $sink
$ns set fid_ 1

```

Por fim é anexada a aplicação FTP ao agente TCP.

```

# Configuração do FTP sobre a ligação TCP
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ftp set type_ FTP

```

O processo descrito em cima é similar para o tráfego UDP. Neste caso considera-se que o tráfego UDP é transferido sobre as duas ligações existentes entre o n0 e o n3 e entre o n1 e o n3.

```

set udp [new Agent/UDP]
$ns attach-agent $n1 $udp
set null [new Agent/Null]
$ns attach-agent $n3 $null
$ns connect $udp $null
$udp set fid_ 2

set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
$cbr set type_ CBR
$cbr set packet_size_ 1000
$cbr set rate_ 1mb
$cbr set random_ false

```

Os próximos passos são as indicações dadas ao escalonador de eventos para seleccionar os diferentes tempos de entrada e saída de funcionamento dos tráfegos bem como a escalonar a altura em que o procedimento *finish* ocorre.

```

# escalonador de eventos para os agentes CBR e
FTP
$ns at 0.1 "$cbr start"
$ns at 1.0 "$ftp start"
$ns at 4.0 "$cbr stop"
$ns at 4.5 "$ftp stop"

$ns at 5.0 "finish"

```

Depois de todas as configurações estarem efectuadas e os processos de escalonamento e de pós simulação estarem acabados, o único procedimento que falta efectuar é começar a simulação. Para isso usa-se o comando:

```
$ns run
```

4.9. RESUMO

O quarto capítulo introduz a ferramenta de simulação ns-2, uma ferramenta do domínio público que permitir a simulação de grande parte dos protocolos utilizados na Internet tanto em redes cabeadas como em redes sem fios.

Depois de uma breve comparação do ns-2 com outra ferramenta de simulação, OPNET, é feita uma introdução do ns-2 demonstrando também o seu funcionamento geral. O ns é um simulador orientado ao objecto, escrito em linguagem C++, com um interpretador OTcl (*object-oriented Tcl*) como *frontend*. O simulador suporta uma hierarquia de classes em C++ e uma hierarquia de classes similar dentro do interpretador OTcl.

Depois, é discutido o processo de simulação onde são demonstrados os aspectos mais importantes na criação de uma topologia de rede para simulação com o ns-2 onde se destacam componentes como o escalonador de eventos, nós, filas, ligações e pacotes ns.

É ainda tema deste quarto capítulo os mecanismos de análise de resultados como o *trace file*, a ferramenta NAM (*Network Animator*) e a linguagem de *script* AWK. Por fim, é feita a demonstração e explicação de um exemplo de configuração de um cenário de simulação.

5. DESCRIÇÃO DOS MÓDULOS DE WIMAX PARA O NS-2

Ao longo do processo de estudo da tecnologia WIMAX foram disponibilizados três módulos para simulação de redes IEEE 802.16 usando o ns-2. Cada um destes módulos integra funções da camada MAC da tecnologia WIMAX seguindo diferentes abordagens para modelização da camada física. Para a simulação de cada módulo foi necessário efectuar implementações diferentes, no ns-2, de forma a que estes se tornassem funcionais. Tendo cada módulo estruturas diferentes e modos de operação distintos, foi necessário efectuar um estudo aprofundado de cada um para serem produzidas as simulações adequadas e, consequentemente, os resultados exactos.

Um dos módulos considerados, implementado pelo *National Institute of Standards and Technology* (NIST), fornece, entre outras funcionalidades, a camada física OFDM com modulação configurável, *Time Division Duplexing* (TDD), topologia ponto-multiponto (*Point-to-Multipoint* - PMP) e fragmentação de pacotes. No entanto, este modelo não

implementa os mecanismos para suporte de QoS na camada MAC, isto é, os diferentes tipos de serviço e mecanismos de escalonamento com QoS.

Outro módulo estudado, desenvolvido pelo *Networks & Distributed Systems Laboratory* (NDSL) também implementa o modo TDD e a topologia PMP utilizando a implementação do canal sem fio já fornecida pelo ns-2. Este modelo já é capaz de implementar os cinco tipos de serviço definidos pela norma 802.16.

O último módulo a ser disponibilizado ao público foi o desenvolvido pela UNICAMP, Universidade Estadual de Campinas, Brasil. Tal como no módulo da NDSL este fornece um mecanismo de escalonamento para o tráfego *uplink* contemplando cada uma das cinco classes previstas pela norma. Este módulo segue as especificações da norma IEEE 802.16 para a topologia PMP e o modo de acesso utilizado é o TDD.

Neste capítulo será feita uma descrição destes três módulos da norma 802.16 para o ns-2.

A geração de cenários é diferente para cada módulo, fazendo com que a programação de cada cenário seja distinta.

5.1. MÓDULO NDSL

5.1.1. INTRODUÇÃO

Este módulo foi desenvolvido por membros do *Networks & Distributed Systems Laboratory* (NDSL) sob a orientação do professor Jenhui Chen. Até a altura da publicação do módulo da NDSL nenhum outro módulo do WIMAX tinha sido apresentado para a ferramenta ns-2. Isto motivou os investigadores da NDSL a contribuírem para a projecção e implementação deste novo módulo [33].

O módulo do WIMAX criado está focado no protocolo da camada MAC já incluído no ns-2. Este módulo é baseado no modo PMP do IEEE 802.16, significando que uma BS é capaz de servir múltiplas SSs em simultâneo. Os investigadores optaram por um esquema OFDMA no que diz respeito à camada física, devido às suas características, uma vez que este esquema demonstra uma grande apetência para as aplicações *wireless*. Esta aptidão é devida à capacidade de atingir altas taxas de transmissão de dados e à sua robustez na propagação de atrasos em multi-caminhos.

5.1.2. ARQUITECTURA IEEE 802.16 DO PONTO DE VISTA DOS AUTORES

Para que seja possível entender-se o funcionamento do módulo é necessário compreender a arquitectura da norma IEEE 802.16 pela vista pelos autores.

Este subcapítulo sumariza as operações entre a camada MAC e a camada física. A *convergence sublayer* (CS) assegura qualquer transformação ou mapeamento de dados da rede externa onde estes tenham sido recebidos através do ponto de acesso ao serviço (SAP – *Service Access Point*) da CS convertendo-os em *MAC Service Data Units* (MSDUs) recebidos na camada MAC através MAC SAP. Esta subcamada também inclui uma classificação de SDUs da rede externa associando-os a um identificador, *MAC service flow identifier* (SFID), e um identificador da ligação (CID – *connection ID*).

A *MAC common part sublayer* (CPS) assegura o núcleo das funcionalidades da camada MAC de acesso ao sistema, alocação da largura de banda, escalonamento, mecanismos de contenção, estabelecimento de ligações e ainda a manutenção das mesmas ligações. A CPS recebe dados provenientes de vários CSs pelo MAC SAP, sendo classificado numa particular ligação MAC.

Como foi visto no capítulo 2 a norma IEEE 802.16 suporta cinco tipos de gestão de QoS: UGS, ertPS, rtPS, nrtPS e BE. Estas classes de QoS estão associadas com certos parâmetros predefinidos na norma. De acordo com estes parâmetros necessários ao bom funcionamento de cada classe, o escalonador MAC suporta os mecanismos apropriados para a manipulação dos dados transportados.

Os *Protocol Data Units* (PDUs) da camada superior são inseridos em diferentes níveis de filas com um CID na camada MAC após o mapeamento SFID-CID. Estes pacotes de dados nestas filas são tratados como MSDUs sendo depois fragmentados ou empacotados em diferentes tamanhos de acordo com as operações do escalonador MAC. Serão processados por um mecanismo de *automatic repeat request* (ARQ).

Para o tráfego UL, cada SS deve “procurar” uma BS antes de entrar no sistema. Este processo é definido como um método que determina a distância da posição da SS até à posição da BS, denominado de *ranging process*. Durante o período inicial desta procura, a SS irá fazer um pedido à BS para ser servida via DL transmitindo as suas opções usando um código *DL interval usage code* (DIUC). Posteriormente, a BS irá comandar a SS para usar um perfil particular de UL usando um código *UL interval usage code* (UIUC) nas

concessões da SS em mensagens UL-MAP. O DL-MAP e o UL-MAP contêm a identificação do canal e os elementos de informação (IEs – *information elements*) MAP que descrevem as especificações físicas mapeadas no UL e DL, respectivamente. Estes são baseados nas diferentes especificações físicas, como as portadoras únicas (SC – *single carrier*), acesso SC (SCa), OFDM e OFDMA.

A camada física requer um equivalente *radio link control* (RLC), que representa a capacidade da camada física transmitir de um *burst profile* para outro.

A norma IEEE 802.16 utiliza uma arquitectura baseada na transmissão de *frames* onde o tamanho deste é variável. Cada *frame* é chamado de *superframe* e é dividido em dois *subframes*: o DL *subframe* e o UL *subframe*, como já foi descrito no capítulo 2.

Apesar da norma IEEE 802.16 definir a sinalização das ligações entre SS e BS, esta não define o processo de controlo de admissão. Todos os pacotes provenientes da camada de aplicação são classificados pelo classificador de ligações baseado no CID sendo depois encaminhados para a fila apropriada. Na SS, o escalonador irá recuperar os pacotes das filas e transmiti-los para a rede nos *time slots* apropriados definidos pela UL-MAP enviado pela BS. A UL-MAP é determinada pelo módulo de escalonamento baseado nas mensagens BW-request. Estas mensagens reportam o tamanho da fila corrente de cada ligação na SS.

5.1.3. DESCRIÇÃO DO MÓDULO 802.16

Segundo os autores o módulo desenvolvido pela NDSL está de acordo com as especificações da norma IEEE 802.16, e baseia-se na versão 2.29 da ferramenta de simulação ns-2. Todos os módulos foram desenvolvidos utilizando a linguagem de programação orientada ao objecto C++ e modulados em diversas classes.

A Figura 52 demonstra a relação entre o módulo do WIMAX e os módulos do ns. Ilustram-se os objectos para a geração de tráfego (TGA – *traffic generating agent*), a camada da ligação (LL – *link layer*), a fila de interface (IFQ – *interface queue*), a camada MAC desenvolvida (módulo do WIMAX) e a camada física.

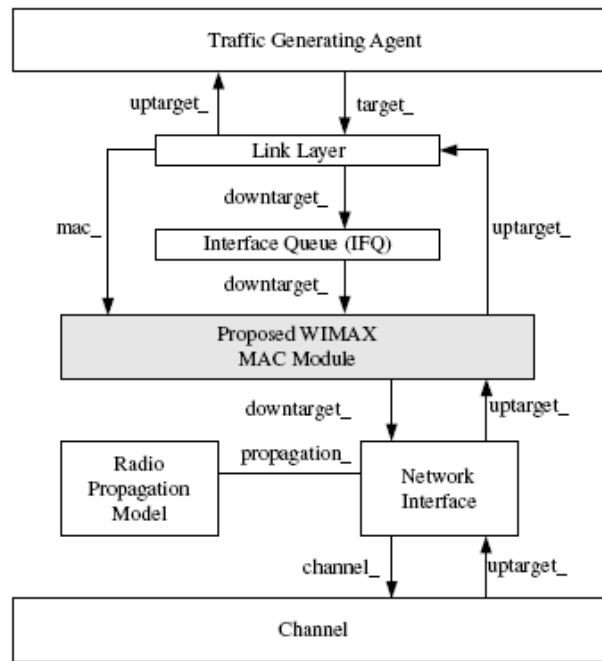


Figura 52 - Relação entre o módulo WIMAX e os módulos do ns [33]

O TGA corresponde à camada de aplicação onde é gerado o tráfego tal como VoIP, MPEG, FTP, tráfego http, etc. Este tráfego é classificado em cinco diferentes tipo de serviço, o UGS, rtPS, ertPS, nrtPS e BE, cada um com o seu nível de prioridade. Todos os pacotes serão transferidos para diferentes tipos de filas com prioridades diferentes de acordo com o seu tipo de serviço, usando, para esse efeito, o mecanismo de mapeamento SFID-CID da camada CS. Os pacotes de dados nestas filas são tratados como MSDUs e será feita uma selecção passando pelo módulo WIMAX num método *round-robin*.

O módulo do WIMAX proposto é composto pela camada CS, pela camada MAC CPS e a camada física. A Figura 53 representa a descrição da arquitectura do módulo 802.16 proposto pelos autores.

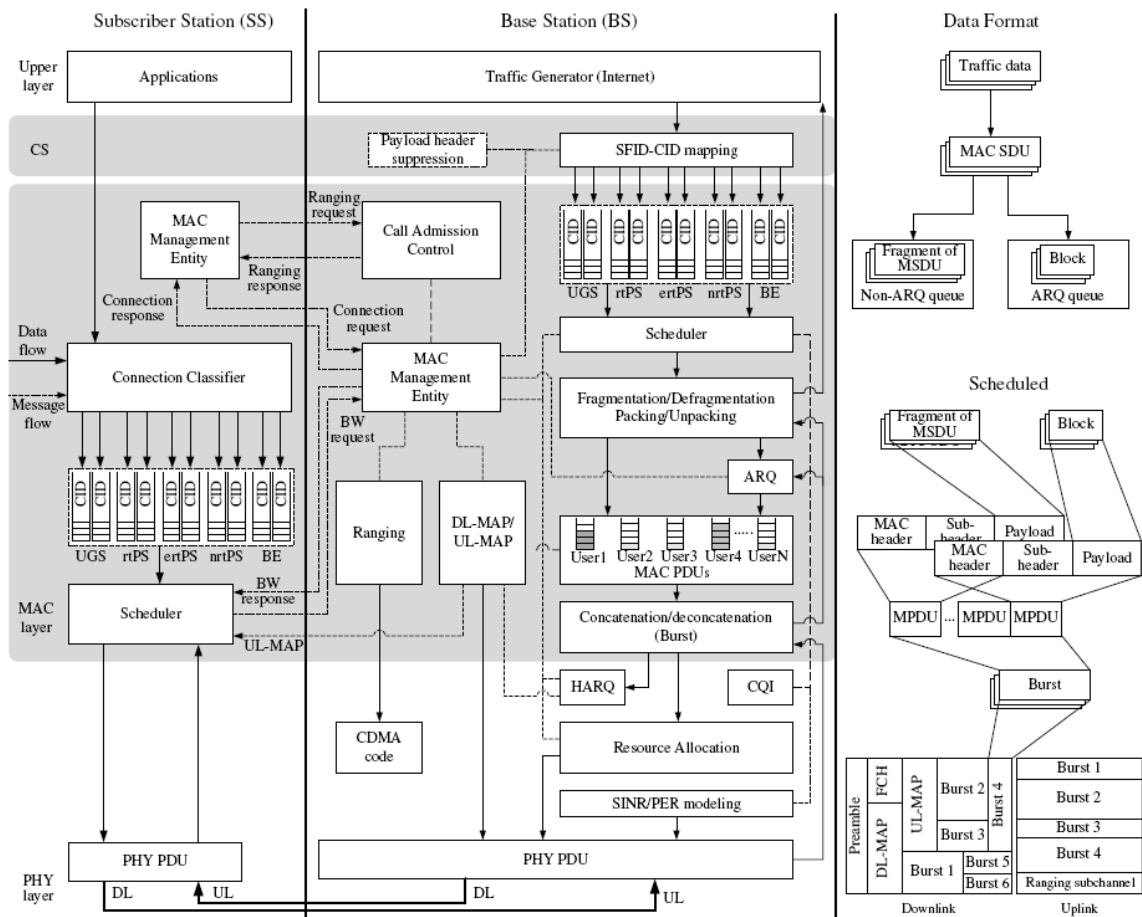


Figura 53 - Arquitetura da camada MAC simulada da norma IEEE 802.16 [33]

A camada CS tem duas funções principais [34]:

1. Transformar o endereço IP, da camada superior, em diversos SFIDs ou o inverso (SFIDs em endereços IP)
2. Gravar o mapeamento entre um SFID e um CID de transporte (TCID)

Estas funções permitem à camada MAC manter a informação essencial da camada superior em relação aos seus parâmetros QoS e endereços de destino.

- Mapeamento IP-SFID
- aceita PDUs provenientes da camada superior;
- efectua uma classificação de PDUs das camadas superiores;

- o SFID é usado tanto para transmissões DL com referências de parâmetros de QoS como para transmissões UL com IP *lookup*;
- entrega CS PDUs ao apropriado MAC SAP.
- Mapeamento SFID-TCID (quando uma SS faz um pedido)
- o mapeamento SFID-TCID é a função principal da camada CS para SFID definindo as classes de serviço de QoS associada a cada ligação;
- as ligações do WIMAX são bidireccionais e cada direcção tem pelo menos cinco prioridades (UGS, rtPS, ertPS, nrtPS, BE).

Gestão da camada MAC [34]:

- Existem cinco tipos de mensagens, DCD, UCD, DL-MAP, UL-MAP e *frames* de pedidos de largura de banda (BR – *bandwidth request*). Todas estas mensagens MAC são desencadeadas por *timers* específicos.
- *Ranging*
 - existem 3 tipos de *ranging*, o inicial, o periódico e o de pedido de largura de banda;
 - segue o mecanismo CSMA e o mecanismo *random back-off*;
- Filas com prioridades
 - tanto na BS como na SS, os pacotes que chegam vindos da camada superior serão entregues consoante a sua prioridade;
 - de acordo com o TCID e o tipo de serviço: UGS(5), rtPS(4), ertPS(3), nrtPS(2), BE(1), será feita uma classificação correspondendo à prioridade.
- Fragmentação / Empacotamento
 - a fragmentação de pacotes ou o processo de empacotamento é executado numa função criada, *PDU_Generator()*;

- esta função recolhe os MSDUs das filas de QoS e produz MPDUs dependendo do comando da função de escalonamento;
- irá gerar um cabeçalho MAC genérico para cada *payload* de dados.

5.1.4. MECANISMOS DE GESTÃO DE QoS

O módulo NDSL é capaz de efectuar a gestão de QoS. Com este efeito, o módulo apresenta um escalonador de tráfego de *uplink* provenientes das diferentes SSs. A política de escalonamento utilizada é o *weighted Round-Robin*. Este método é uma versão melhorada do método *Round-Robin* que elimina algumas limitações deste algoritmo.

O *Round-Robin* é um algoritmo simples e rápido de escalonamento, ideal para sistemas onde não há muito tempo para tomar decisões. Uma pequena unidade tempo é denominada por *quantum* ou *timeslice*. Neste caso todos os processos são armazenados numa fila circular. O escalonador percorre a fila, alocando o processador para cada procedimento durante um *quantum*. Assim, o escalonador retira o processo da fila e dedica-lhe um *quantum*. Caso o processo não tenha acabado neste período de tempo o escalonador coloca-o no fim da fila.

A grande diferença entre o método *Round-Robin* e o *weighted Round-Robin* é que no caso do algoritmo *weighted Round-Robin* é atribuído um peso a cada processo podendo-se definir prioridades a cada tipo de processo. Neste método as ligações podem ser tratadas como uma sessão separada. Assim para o caso do WIMAX este algoritmo é capaz de atribuir diferentes prioridades a cada tipo de classe, fornecendo quantidades maiores de tempo a classes que possuam requisitos mais elevados.

Exemplo da distribuição de pesos pelas classes de tráfego usado no método *weighted Round-Robin* para efectuar o escalonamento das diferentes classes[34].

$$2^{\circ} \text{ round:} \quad 10\% + 30\% + 15\% + 0\% + 0\% = 55\%$$

$$1^{\circ} \text{ round:} \quad 10\% + 10\% + 10\% + 10\% + 10\% = 50\%$$



Figura 54 - Esquema do método Weighted Round-Robin [34]

No primeiro *round* é atribuída uma percentagem igual a cada tipo de classe. Com isto, é possível garantir que tráfegos de prioridades inferiores consigam obter alguma largura de banda. No segundo *round*, o escalonador irá servir os restantes serviços ainda inacabados numa ordem de prioridades. Este processo irá ser repetido até que a largura de banda se esgote ou os restantes serviços sejam completamente servidos. Como se pode analisar (Figura 54), não é possível atribuir as percentagens correctas a cada classe, tornando este método de escalonamento pouco eficaz.

Os criadores do módulo consideram que este método de escalonamento não é a melhor opção para escalonar o tráfego uplink para a tecnologia WIMAX, prometendo substituir este método por um mais eficaz em trabalhos futuros.

5.1.5. IMPLEMENTAÇÃO

O módulo criado pela NDSL foi optimizado para a versão 2.29 da ferramenta ns-2. Sendo esta versão, uma versão já desactualizada, foi feito o esforço de implementar este módulo na versão mais recente, até à data, o ns-2.32.

Depois da descompactação da última versão do módulo da NDSL, versão wimax_v2.03 [35], verifica-se três pastas (*mac*, *common* e *queue*). Dentro das 3 pastas estão incluídos os novos ficheiros que serão necessários adicionar ao ns-2 para que possa ser utilizado o módulo. Dentro da pasta *mac* está a subpasta *mac-802.16* onde se encontra o código em C++ com o comportamento do módulo. Nesta subpasta é introduzida uma pasta chamada *traffic* que contém o código respeitante às 5 classes de QoS disponíveis para o WIMAX. Depois de alterar o ficheiro *Makefile*, para que este compile os novos ficheiros, o módulo deveria estar preparado para ser executado, o que não foi o caso.

Para que o módulo funcionasse foi necessário efectuar algumas alterações nos novos ficheiros introduzidos, de forma a ficarem compatíveis com o ns-2 actualizado. A partir da versão 2.30 foi acrescentado ao ns um novo módulo denominado *PackMimeHTTP*. Este é

um novo gerador de tráfego implementado na *Bell Labs*. Como consequência desta nova introdução, ficheiros como o `packet.h` do ns foram alterados nas novas versões, adicionando, por exemplo, um novo tipo de pacote, `PT_BLTRACE`, e as suas configurações. Ao instalar o módulo do WIMAX criado pela NDSL foram substituídos vários ficheiros, sendo um deles, o `packet.h`. O novo ficheiro introduzido apresenta as configurações encontradas na versão 2.29, sendo necessário, por tanto, acrescentar as alterações adicionadas nas versões posteriores do ns.

Outra das principais actualizações, entre estas duas versões do ns-2, foi a alteração feita nos mecanismos emissão do ficheiro de monitorização de todo o tráfego efectuado pela totalidade dos pacotes que percorrem a rede, o *trace file*. Com este intuito foram realizados inúmeros testes, assim como várias *scripts* até atingir os resultados correctos.

Depois de estes processos terem sido concluídos foi possível realizar as experiências.

5.1.6. EXEMPLO DE UM CENÁRIO DE SIMULAÇÃO DEFINIDO PARA O MÓDULO NDSL

O processo de criação de um cenário de simulação com o módulo WIMAX da NDSL implica a construção de uma *script* Tcl que configure e invoque as funções implementadas.

Como é comum na geração de cenários em ambientes wireless a *script* Tcl começa com a introdução de especificações referentes a várias opções.

```
set val(chan) Channel/WirelessChannel
set val(prop) Propagation/TwoRayGround
set val(netif) Phy/WirelessPhy
set val(mac) Mac/802_16
set val(ifq) Queue/DropTail/PriQueue ;#tipo
de fila
set val(ll) LL ;#tipo de ligação
set val(ant) Antenna/OmniAntenna
antenna model
set val(ifqlen) 50
set val(nn) 6 ;# número de nós
set val(rp) DSDV ;# protocolo de routing
set val(x) 1000
set val(y) 1000
set val(stop) 15.0 ;# tempo do fim da simul.
```

Entre estas opções destacam-se a escolha do canal utilizado, como sendo wireless assim como a opção de se utilizar o MAC introduzido no ns-2 após a implementação do módulo. São definidas também opções como o número de nós introduzidos, o tipo de protocolo de *routing* utilizado, que no caso é DSDV em que neste tipo de router poderá existir trocas de

mensagens entre nós vizinhos. É ainda introduzido o tempo final da simulação que neste caso é de cinco segundos.

A definição do cenário com este módulo apresenta os vários passos já mencionados no capítulo da ferramenta de simulação, como criações de objectos de simulação, nós e ligações.

A configuração dos nós é feita de uma forma genérica envolvendo as variáveis e opções configuradas no extracto de código atrás apresentadas. Como é ilustrado no seguinte extracto de código.

```
$ns node-config -adhocRouting $val(rp) \  
               -llType $val(ll) \  
               -macType $val(mac) \  
               -ifqType $val(ifq) \  
               -ifqLen $val(ifqlen) \  
               -antType $val(ant) \  
               -propType $val(prop) \  
               -phyType $val(netif) \  
               -channel $chan0 \  
               -topoInstance $topo \  
               -agentTrace ON \  
               -routerTrace ON \  
               -macTrace ON \  
               -movementTrace ON
```

As últimas linhas do extracto anterior encontradas referem-se à activação da monitorização do tráfego a diferentes níveis (ao nível do agente, do router, do MAC e movimentos).

O seguinte extracto reporta-se a um exemplo da criação de uma ligação com o protocolo de transporte UDP inicializando uma aplicação CBR incluída na classe UGS.

```
set udp0 [new Agent/UDP]  
$ns attach-agent $node_(1) $udp0  
set null0 [new Agent/Null]  
$ns attach-agent $node_(0) $null0  
$ns connect $udp0 $null0  
$udp0 set fid_ 2  
set cbr0 [new Application/Traffic/UGS]  
$cbr0 attach-agent $udp0  
$cbr0 set type_ UGS  
$cbr0 set packet_size_ 1000  
$cbr0 set rate_ 512Kb  
$ns at 0.0 "$cbr0 start"  
$ns at 10.0 "$cbr0 stop"
```

O módulo implementado pela NDSL possui na pasta mac uma subpasta de classes de tráfego, onde cada uma apresenta as características definidas para cada tipo de fluxo.

Assim, ao anexar o objecto CBR à ligação UDP indica-se que a ligação entre o nó 1 (SS) e o nó 0 (BS) será pertencente à classe UGS. O procedimento encontrado nestas linhas de código é repetido para cada estação introduzida, tornando este processo um pouco moroso.

A estrutura do código Tcl no módulo NDSL acaba com a introdução do seguinte extracto de código:

```
$ns at $val(stop) "$ns nam-end-wireless
$val(stop)"
$ns at $val(stop) "stop"
$ns at 15.1 "puts \"end simulation\" ; $ns halt"

proc stop {} {
    global ns tracefd namtrace
    $ns flush-trace
    close $tracefd
    close $namtrace
}
$ns run
```

É possível verificar que são terminados os processos de monitorização dos pacotes tanto ao nível da ferramenta NAM como do *trace file* introduzindo também o tempo em que a simulação acaba.

5.2. MÓDULO NIST

5.2.1. INTRODUÇÃO

O módulo implementado pelo *National Institute of Standards and Technology* (NIST) [36] é baseado na norma IEEE 802.16-2004 e na sua extensão 802.16e. A Tabela 7 lista as funcionalidades do módulo bem como algumas das suas limitações.

Tabela 7 - Funcionalidades / limitações do módulo da NIST

Funcionalidades disponíveis	Funcionalidades NÃO implementadas
<p>Camada física WirelessMAN-OFDM com modulação configurável</p> <p>Time Division Duplexing (TDD)</p> <p>Mensagens de controlo para executar entradas na rede (sem autenticação)</p> <p>Escalonador fornece uma alocação de uplink pelo método round robin registando as MSs de acordo com a largura de banda requerida</p> <p>Extensão IEEE 802.16e para suportar pesquisas e handovers</p> <p>Fragmentação e reassemblagem de frames</p>	<p>WirelessMAN-OFDMA</p> <p>Frequency Division Duplexing (FDD)</p> <p>ARQ (Automatic Repeat Request)</p> <p>Escalonamento de QoS</p> <p>Ranging periódico e ajustes de energia</p> <p>Empacotamento</p> <p>Correcção de erros</p>

5.2.2. VISTA GERAL DO MÓDULO

O módulo desenvolvido pela NIST inclui a camada MAC bem como a subcamada superior CS para além de modulação OFDM para a camada física.

CAMADA CS

A camada CS reside em cima da camada MAC executando as seguintes funções:

- Recebe PDUs de camadas superiores;
- Executa a classificação;
- Entrega os PDUs do CS ao SAP da camada MAC;
- Recebe CS PDUs de outras entidades (*peer*).

No módulo implementado, a camada CS apenas executa a classificação. A Figura 55 ilustra a estrutura do classificador.

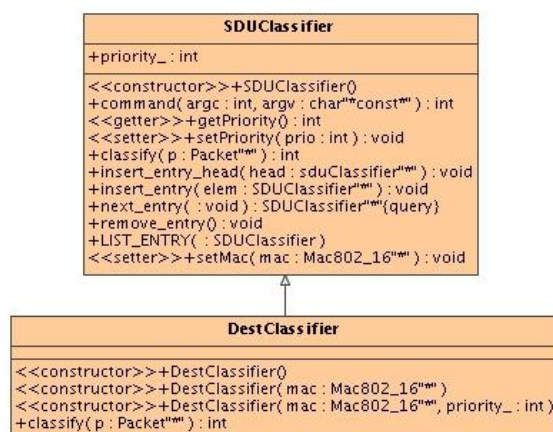


Figura 55 - Diagrama da classe *Classifier* [36]

O *SDUClassifier* constitui um método de definição de prioridade, podendo ser usado para especificar a ordem com que cada classificador é chamado. Quanto menor for o número da prioridade mais depressa o classificador será chamado.

A classe *DestClassifier* usa o endereço MAC de destino localizado no pacote e o tipo de pacote para determinar o CID apropriado.

CAMADA MAC

A camada MAC deste módulo suporta a topologia PMP.

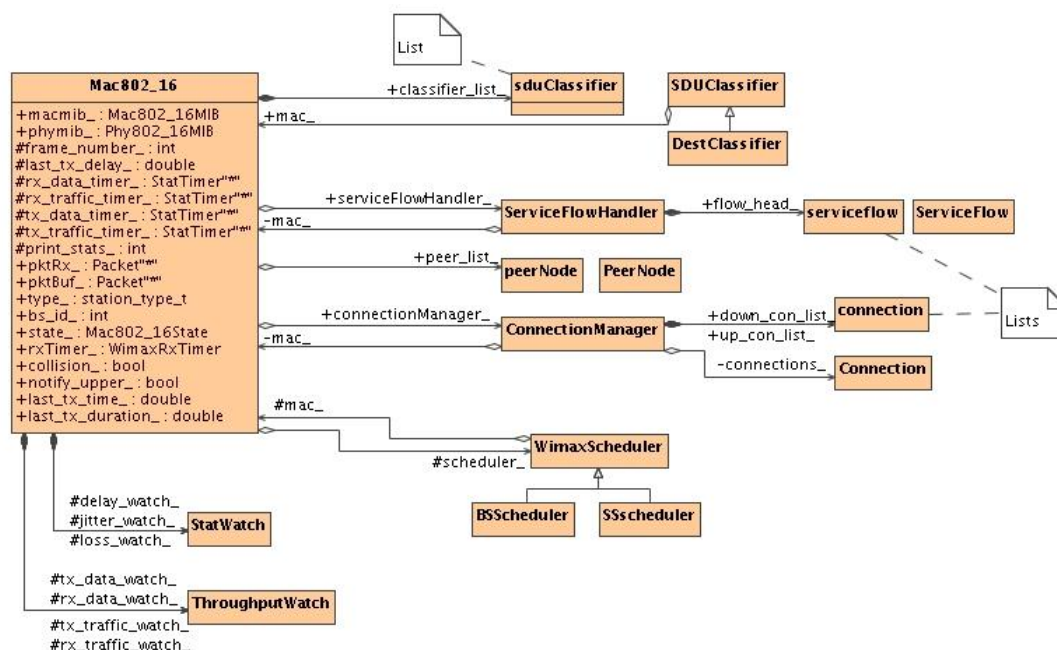


Figura 56 - Diagrama da classe MAC 802.16 [36]

Mac802_16 é uma subclasse da classe MAC. É uma classe abstracta que contém os elementos comuns da BS e das MSs. A classe *Mac802_16* funciona como o interface com outras camadas para o envio e recepção de pacotes. A Figura 56 apresenta a classe bem como a sua relação com os outros módulos.

O módulo *ServiceFlowHandler* é responsável pelo manuseamento dos pedidos e respostas de fluxo armazenando também a lista de fluxos de cada nó.

Uma SS é registada numa BS, podendo a BS estar conectada com múltiplas SSs. A classe *PeerNode* contém informações sobre o *peer* tais como as suas ligações e o seu status. As ligações estão acessíveis através da classe *ConnectionManager* contendo a lista de ligações de entrada e de saída.

A classe abstracta *WimaxScheduler* é usada para criar um interface com o MAC. Existem dois tipos principais de escalonadores: um para a BS e outro para a SS.

Endereçamentos e ligações

Cada MAC possui um código de endereçamento que está definido na classe MAC do ns-2. O módulo define também identificadores de ligações (CIDs).

Durante a inicialização, na BS, são criadas as seguintes ligações:

- *Ranging* inicial (entrada e saída);
- Enchimento (entrada e saída);
- Broadcast (saída);
- Adaptative Antenna System (AAS).

Durante a inicialização, na SS, são criadas as seguintes ligações:

- *Ranging* inicial (entrada e saída);
- Enchimento (entrada e saída);
- Broadcast (saída).

Adicionalmente, durante a entrada na rede, são atribuídos os seguintes CIDs:

- CID básico (entrada e saída);
- CID primário (entrada e saída);
- CID secundário (entrada e saída);
- CIDs de dados.

Mensagens de controlo

O módulo WIMAX da NIST define diversas mensagens de controlo, tais como mensagens de sincronização, mensagens de controlo de fluxos de serviço e mensagens de controlo de mobilidade. A Tabela 8 sumaria as diversas mensagens de controlo definidas:

Tabela 8 - Mensagens de controlo definidas

Categoria	Mensagens definidas
Sincronização	DL-MAP / DCD UL-MAP / UCD RNG-REQ/RSP REG-REQ/RSP
Fluxos de serviço	DSA-REQ/RSP/ACK
Mobilidade	MOB_NBR_ADV MOB_SCN-REQ/RSP MOB_BSHO-REQ/RSP MON_SSHO-REQ MOB_HO-IND MOB_SCN-REP MOB_ASC-REP

Fragmentação

O módulo da NIST é capaz de realizar a fragmentação. Quando é efectuado o escalonamento de pacotes para transmissão, o escalonador verifica se a fragmentação está activa para a ligação dividindo o pacote para caber no *burst*.

Inicialização e entrada na rede

Sempre que uma SS deseja juntar-se a uma rede, tem que executar uma entrada na rede. Como se ilustra na Figura 57, o módulo implementa as seguintes funcionalidades para a entrada na rede:

- Procura do canal de *downlink*;
- Obtenção dos parâmetros de transmissão;
- *Ranging* inicial;
- Registo.

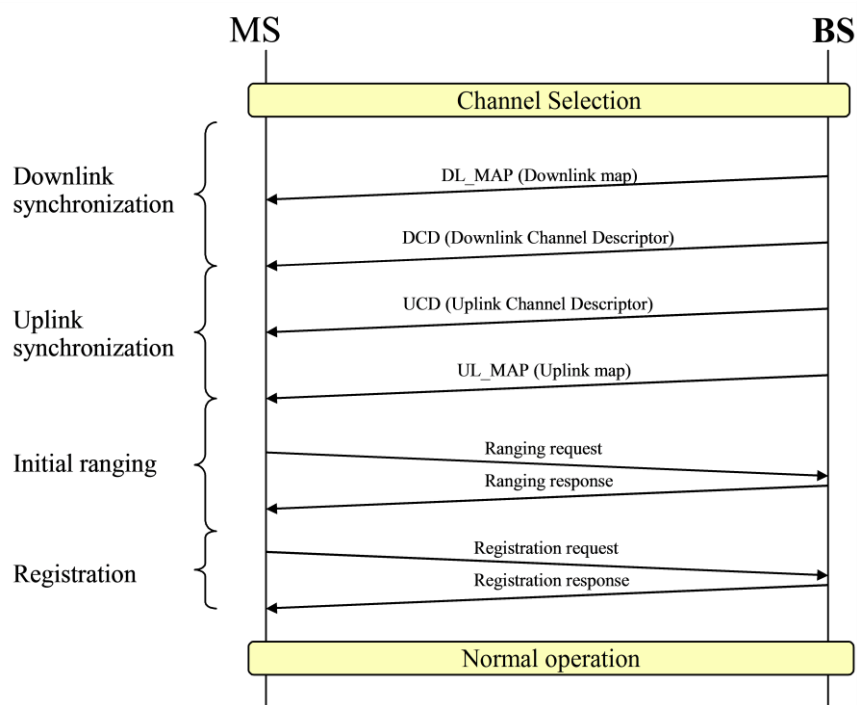


Figura 57 Entrada na rede [36]

Bem como a configuração de alguns parâmetros:

- Temporizadores que executam a procura do canal;
- Frequência das mensagens DCD/UCD;
- Parâmetros para o *ranging* inicial;
- Alocação do canal.

Procedimentos de *handover* da camada MAC

Como já foi referido este módulo suporta mobilidade. Dependendo da sua configuração, uma SS pode efectuar *scanning* e *handover* entre BSs.

Quando a qualidade de uma ligação começa a deteriorar-se uma MS pode enviar um pedido MOB_SCN_REQ à BS que a está a servir pedindo um intervalo de *scanning* com o objectivo de encontrar BSs circundantes.

Durante o processo de *scan* a MS recolhe valores RSSI (*Received Signal Strength Indication*) dos pacotes recebidos. Estes valores são reportados à BS que a está a servir. A BS usa estes valores para seleccionar a nova BS.

Após a MS receber a indicação da BS seleccionada, espera alguns *frames* antes de indicar as suas intenções de efectuar o *handover*.

A Figura 58 demonstra a sequência de mensagens durante o *scanning*.

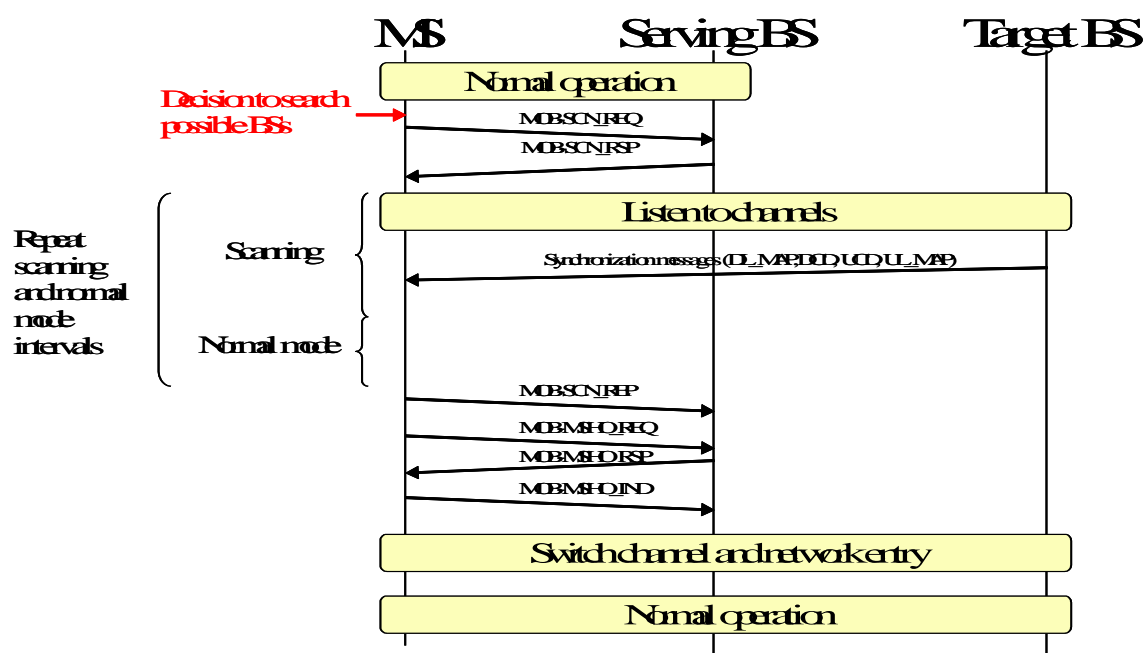


Figura 58 - Processo de *scanning* [36]

Processamento dos pacotes

O diagrama apresentado na Figura 59 demonstra de uma forma mais detalhada como um pacote percorre a camada MAC.

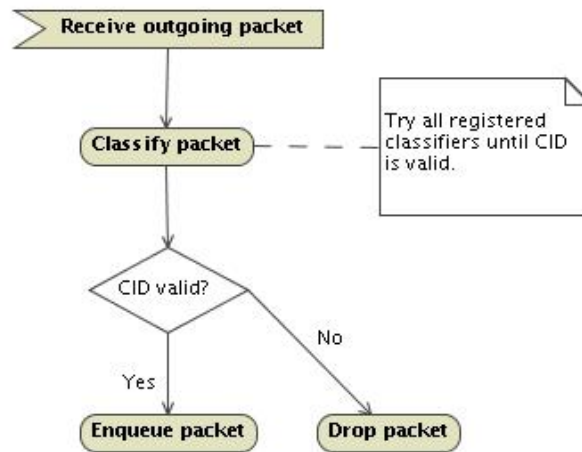


Figura 59 - Processamento dos pacotes de saída [36]

Um pacote recebido de uma camada superior é classificado usando os classificadores registados. Uma vez que pode haver vários classificadores, a camada MAC analisa-os, um a um, até que um CID válido seja encontrado. Se o CID é válido, o pacote é adicionado à fila correspondente, se não, o pacote é descartado.

Camada física

A camada física OFDM é usada para transmitir pacotes no módulo implementado. A camada física pode estar em diferentes estados. Quando está no modo de envio, todos os pacotes recém-chegados são descartados. No modo de recepção, os pacotes não podem ser enviados. Além disso, o cabeçalho do pacote contém informação virtual, tal como, frequência, modulação e um prefixo cíclico, que são utilizados para filtrar os pacotes recém-chegados.

O módulo suporta diferentes tipos de modulação. A camada MAC atribui *bursts* que podem utilizar diferentes modulações de acordo com a distância ou com interferências. Isto afecta a taxa de transferência de dados e tempos de transmissão [36].

5.2.3. MECANISMOS DE GESTÃO DAS CLASSES DE QOS

Neste módulo, o escalonador é incapaz de efectuar a atribuição de prioridades relativamente a cada uma das classes, obrigando a que cada classe seja tratada da mesma forma. Este escalonador interage com o MAC através de uma API permitindo configurar diversas implementações.

As estruturas de classes do módulo permitem especificar diferentes serviços de dados, nomeadamente UGS, rtPS, nrtPS e BE.

Diferentes tipos de nós requerem diferentes tipos de serviços. Na norma 802.16, a BS controla a alocação da largura de banda existindo um número elevado de possibilidades distintas de implementação. Este módulo inclui uma classe abstracta, *WimaxScheduler* capaz de atribuir a cada nó uma classe de serviço diferente, não atribuindo a cada uma um grau de prioridade. Para isso, é necessário o utilizador especificar a classe a utilizar usando *script* TCL.

5.2.4. IMPLEMENTAÇÃO

O módulo criado pela NIST foi criado para a ferramenta de simulação ns versão 2.29. Comparando com os outros dois módulos testados neste trabalho, este faz-se acompanhado por um patch que permite a execução completa da sua instalação para a versão 2.29 e para versões superiores. Com isto é possível instalar o módulo na versão 2.32 (utilizada na realização do trabalho), sem que se depare com erros de instalação que necessitem de ser corrigidos.

A versão instalada do módulo foi a ultima disponibilizada, a versão prerelease041707.

5.2.5. EXEMPLO DE UM CENÁRIO DE SIMULAÇÃO COM O MÓDULO NIST

Os excertos de código seguidamente apresentados têm o objectivo de explicar o processo de geração de cenários de simulação utilizando o módulo NIST.

É importante referir que este exemplo utiliza um cenário composto por uma BS rodeada por um número de SSs móveis (MSs) num ambiente wireless. Esta BS está também ligada através de uma ligação por cabo, a um nó *sink* simulando a ligação da BS à internet. Neste tipo de cenários é comum ser utilizado um roteamento feito de uma forma hierárquica.

No início do ficheiro de geração de um cenário pode ser feita a introdução das variáveis globais.

```
# set global variables
set nb_mn [lindex $argv 0];# Max n° de MSs
set packet_size 1500 ;# tamanho dos pacotes
set output_dir .
set gap_size 1
puts "gap size=$gap_size"
set traffic_start 20
```



```
set traffic_stop 120
set simulation_stop 140
set diuc 7 ;# Modulação para MNs
```

Aqui são introduzidos valores como o tempo de simulação, o tamanho dos pacotes, os instantes de início e fim da geração do tráfego e ainda o número máximo de nós móveis que são introduzidos pelo utilizador através da linha de comando.

As configurações das opções iniciais são feitas de uma forma similar ao efectuado pelo módulo NDSL. Neste caso interessa mostrar a escolha da camada física e da camada MAC utilizadas.

```
Phy/WirelessPhy/OFDM set g_ 0.25
set opt(mac) Mac/802_16/BS
```

As seguintes linhas de código referem-se à configuração do router de uma forma hierárquica.

```
$ns node-config -addressType hierarchical
AddrParams set domain_num_ 2
lappend cluster_num 1 1
AddrParams set cluster_num_ $cluster_num
lappend eilastlevel 1 [expr ($nb_mn+1)]
AddrParams set nodes_num_ $eilastlevel
```

Esta forma de roteamento hierárquico define vários níveis de domínios, adicionando a cada domínio vários agrupamentos de nós. Neste caso são definidos dois domínios sendo introduzido um agrupamento de nós em cada. Depois é acrescentado um número de MSs que pertencerá ao último agrupamento. Os outros agrupamentos apenas irão conter a BS num e o nó sink no outro.

O processo de atribuição das opções, já descritas em cima, é efectuado a dois níveis. Primeiro atribuem-se ao nó da BS as seguintes opções:

```
$ns node-config -adhocRouting $opt(adhocRouting)
\
    -llType $opt(ll) \
    -macType Mac/802_16/BS \
    -ifqType $opt(ifq) \
    -ifqLen $opt(ifqlen) \
    -antType $opt(ant) \
    -propType $opt(prop) \
    -phyType $opt(netif) \
    -channel [new $opt(chan)] \
    -topoInstance $topo \
    -wiredRouting ON \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
```

```
-movementTrace OFF
```

Depois atribui-se às SSs:

```
$ns node-config -macType Mac/802_16/SS \  
                -wiredRouting OFF \  
                -macTrace ON
```

Neste módulo foram criados dois tipos de MAC. um para a BS e outro para as MSs.

A criação dos nós é realizada da seguinte forma

```
set sinkNode [$ns node 0.0.0]  
set bstation [$ns node 1.0.0]
```

No caso da BS e do nó *sink* são introduzidos os valores que definem a sua posição dentro da hierarquia. Depois é configurado o nó para não se movimentar.

```
$bstation random-motion 0
```

No caso dos nós móveis a configuração pode ser feita da seguinte forma, onde é criado cada um dos nós e depois anexado à BS.

```
for {set i 0} {$i < $nb_mn} {incr i} {  
    set wl_node_($i) [$ns node 1.0.[expr $i + 1]]  
  
    $wl_node_($i) base-station [AddrParams addr2id  
    [$bstation node-addr]] ;#attach mn to basestation  
    (...)  
}
```

A restante parte da configuração das ligações de cada nó móvel é efectuada de uma forma similar à apresentada na secção da configuração de cenários de simulação, presente no capítulo 3. É importante referir que cada nó móvel cria uma ligação até ao nó *sink*.

```
# Dentro do ciclo for:  
(...)  
set udp_($i) [new Agent/UDP]  
$udp_($i) set packetSize_ 1500  
$ns attach-agent $wl_node_($i) $udp_($i)  
  
set cbr_($i) [new Application/Traffic/CBR]  
$cbr_($i) set packetSize_ $packet_size  
$cbr_($i) set interval_ $gap_size  
$cbr_($i) attach-agent $udp_($i)  
set null_($i) [new Agent/Null]  
$ns attach-agent $sinkNode $null_($i)  
  
$ns connect $udp_($i) $null_($i)  
(...)
```

Para melhor compreensão do cenário foi produzido o diagrama apresentado na Figura 60.

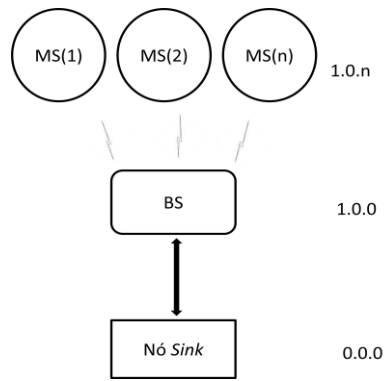


Figura 60 - Topologia do cenário criado

É importante referir que este módulo vem munido de funcionalidades que permitem retirar os valores da taxa de transferência efectiva e de atrasos dos cenários estudados, permitindo uma poupança de tempo enorme quando comparado com os outros dois módulos estudados que necessitam de *scripts* auxiliares para se retirarem os resultados dos seus ficheiros de *output*.

5.3. MÓDULO UNICAMP

5.3.1. INTRODUÇÃO

Este módulo foi desenvolvido por Juliana Freitag Borin e Nelson L. S. da Fonseca da Universidade Estadual de Campinas (UNICAMP), Brasil. O seu desenvolvimento tem como foco a camada de acesso ao meio utilizando a camada física sem fios já disponibilizada no ns-2. A implementação contempla os mecanismos de reserva e alocação de banda e de provisão de QoS especificados na norma IEEE 802.16 [37][38].

O módulo para simulação de redes WIMAX foi desenvolvido na versão 2.28 do simulador ns-2 e segue as especificações da norma IEEE 802.16 para a topologia PMP e o modo de acesso TDD. A implementação foi realizada na linguagem C++ utilizando programação orientada a objectos.

5.3.2. VISTA GERAL

O módulo WIMAX baseou-se num módulo desenvolvido para o ns-2 para simulação da norma DOCSIS, utilizado para transmissão de vídeo em redes cabeadas. A reutilização deste código foi possível pois a especificação da camada MAC da norma IEEE 802.16 é baseada na norma DOCSIS. Assim como na norma IEEE 802.16, no padrão DOCSIS

existe uma entidade central que controla o acesso ao meio fazendo a alocação de banda para as estações cliente. A alocação da banda baseia-se nas requisições enviadas pelas estações clientes e nos requisitos de QoS das ligações.

Embora a camada MAC de ambas as normas seja semelhante, foram efectuadas modificações no código do módulo DOCSIS para adequá-lo à norma IEEE 802.16. Entre essas modificações destacam-se: a implementação do serviço nrtPS para o tráfego no *uplink*, a implementação dos cinco tipos de serviço para o tráfego no *downlink*, a alteração da interface entre a camada MAC e a camada física para que o módulo WIMAX utilizasse o meio sem fios disponível no ns-2, a implementação da divisão do tempo em *frames* e dos *frames* em *subframe downlink* e *subframe uplink* e a inclusão dos parâmetros de QoS latência máxima, para o serviço rtPS, e largura de banda mínima, para os serviços rtPS e nrtPS.

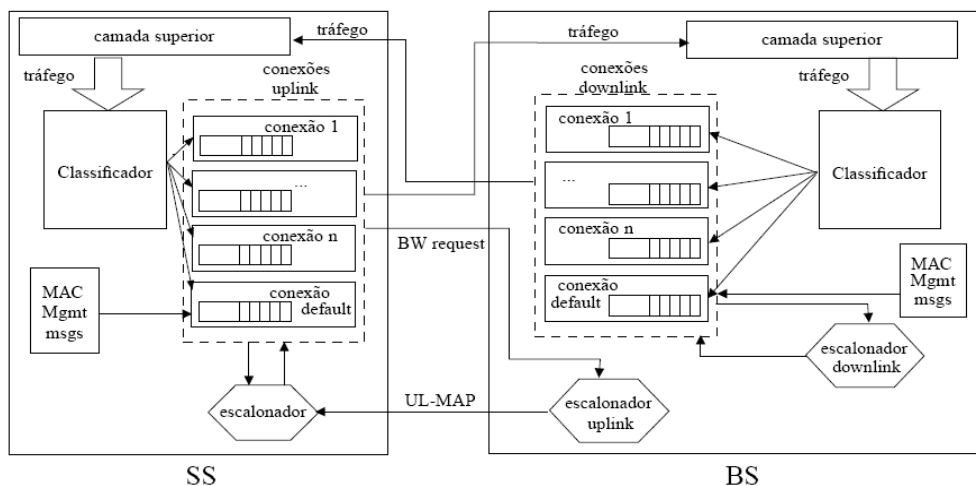


Figura 61 - Estrutura do módulo para a simulação na norma IEEE 802.16 [38]

A Figura 61 apresenta a estrutura do módulo implementado. Tanto na BS como nas SSs o tráfego proveniente da camada superior é classificado para o fluxo de serviço apropriado (associado a uma ligação) a partir do tipo do tráfego e do par origem e destino. Os fluxos de serviço são configurados pelo utilizador, que entre outras informações, determina os requisitos de QoS. Cada nó da rede deve ter uma ligação *default* para transmitir mensagens de gestão e o tráfego que não puder ser mapeado para nenhuma das outras ligações. No início da simulação todas as SSs registam-se na BS para o estabelecimento das ligações, com seus respectivos parâmetros de QoS. A BS atribui um identificador único a cada ligação e armazena diversa informação numa tabela, como por exemplo: o tipo do serviço

e seus parâmetros de QoS, se a ligação utiliza as funcionalidades de fragmentação, concatenação e *piggybacking*, e o identificador único dessa ligação.

A BS possui um escalonador de *uplink* e um escalonador de *downlink*. O escalonador *downlink* determina quais dos pacotes enviados pela camada superior serão transmitidos em cada *subframe downlink*. Para tomar esta decisão, o escalonador baseia-se nos requisitos de QoS e na situação da fila de cada ligação. O escalonador de *uplink* decide quais SSs terão direito de transmitir em cada *subframe uplink* e quantos *slots* cada uma dessas SSs poderá utilizar. A decisão deste escalonador leva em consideração os requisitos de QoS das ligações *uplink*, bem como as requisições de banda recebidas.

Na norma IEEE 802.16, no início de cada *subframe downlink*, a BS envia um mapa *downlink* (DL-MAP) e um mapa *uplink* (UL-MAP) para informar quais SSs receberão pacotes no *subframe downlink*, quais poderão transmitir no *subframe uplink* e o número de *slots* reservados para cada transmissão. No módulo implementado, o DL-MAP não é enviado, dado que as informações que ele conteria não são necessárias num sistema de simulação. No início do *subframe downlink* são reservados alguns *slots* para simular a sobrecarga do envio dessa mensagem. O UL-MAP é enviado e contém a descrição dos *slots* alocados no *subframe uplink* subsequente.

Nas SSs existe um escalonador que decide quais pacotes provenientes da camada superior serão enviados nos *slots* recebidos. Para tomar esta decisão, o escalonador baseia-se nas informações transportadas pelo UL-MAP, nos requisitos de QoS e na situação das filas das ligações.

Na norma IEEE 802.16, os *frames* têm uma duração predefinida, enquanto os *subframes uplink* e *downlink* podem ter tamanho variáveis dependendo da necessidade de cada um. O módulo de simulação permite que o utilizador defina a duração dos *frames*. Dado esse valor, o módulo atribui durações iguais para cada um dos *subframes*.

Num sistema real, a BS e as SSs trocam em torno de 50 tipos de mensagens de gestão, como por exemplo mensagens de *ranging*, mensagens com informações sobre o canal, mensagens para garantir a segurança na rede, entre outras. A maioria destas mensagens, em especial aquelas que envolvem informações sobre o canal, autenticação e segurança na rede, não é implementada por estarem fora dos objectivos do módulo desenvolvido. No

entanto, mensagens de gestão são geradas periodicamente para representar a sobrecarga necessária.

Requisição e alocação de largura de banda

Quando um pacote chega na camada MAC a partir das camadas superiores, ele é classificado para um fluxo de serviço (associado a uma ligação) baseado no endereço IP de origem e destino e no tipo do pacote (HTTP, FTP, CBR, etc.). A seguir, o cabeçalho MAC de 6 bytes é incluído no pacote. Cada fluxo de serviço possui uma fila onde os pacotes são inseridos caso não haja um *grant* disponível. Os pacotes que chegam pelo canal são classificados como uma mensagem de gestão ou de dados através das informações disponíveis no cabeçalho e manipulados adequadamente.

Cada fluxo de serviço possui quatro componentes principais:

- Classificador: usa endereços de origem e destino e o tipo do pacote para classificar um pacote para um fluxo de serviço.
- Fila: todos os pacotes classificados para um fluxo de serviço são inseridos na fila caso não possam ser imediatamente enviados.
- Tabela de Alocação: mantém as informações sobre os *grants* alocados para o fluxo de serviço. Esta tabela é actualizada sempre que um UL-MAP é recebido.
- Máquina de estados finita (FSM – *Finite State Machine*): determina as acções de um fluxo de serviço de acordo com os eventos que ocorrem na rede. Sempre que um evento ocorre, dependendo do estado em que a máquina se encontra, a função apropriada é chamada para tratar o evento de entrada e a variável de estado é actualizada.

A definição das máquinas de estado para os serviços UGS, rtPS e BE no sentido *uplink* foram baseadas nas máquinas de estado propostas em [Shrivastav 2003]. A máquina de estados para o serviço nrtPS utiliza o mesmo modelo da máquina de estados do serviço BE, dado que ambos os serviços utilizam *polling unicast* e de contenção. A diferença está no facto de que a BS aloca *grants unicast* frequentemente para o serviço nrtPS, enquanto o serviço BE recebe *grants unicast* apenas quando há sobra de recursos. A máquina de estado para o serviço ertPS foi definida de forma a permitir o envio tanto de dados como

de requisições de banda nos *grants* alocados periodicamente pela BS. Assim como o serviço UGS no sentido *uplink*, os fluxos de serviço no sentido *downlink* não precisam enviar requisições, sua tarefa consiste em enviar os dados nos *grants* alocados pela BS. Portanto, todos os serviços no sentido *downlink* usam a mesma máquina de estados modelada para o serviço UGS no sentido *uplink*.

5.3.3. MECANISMO DE GESTÃO DE QoS

No módulo implementado pela UNICAMP é proposto um algoritmo para que a BS faça a gestão dos serviços de QoS para o tráfego de *uplink* [39]. De acordo com a norma 802.16 o mecanismo de gestão da BS para o tráfego *uplink* fornece concessões (*time slots*) em intervalos periódicos a fluxos de UGS para enviar dados. Estas concessões periódicas também são fornecidas aos fluxos de rtPS e nrtPS para estes requererem largura de banda. Antes de satisfazer os pedidos de largura de banda, o mecanismo de gestão deve garantir que as exigências da largura de banda e dos atrasos sejam encontrados. A BS executa a gestão do tráfego durante cada trama e faz um *broadcast* da mesma numa mensagem de UL-MAP na sub-trama de *downlink*. Este mecanismo de gestão proposto usa três filas: fila de baixa prioridade, fila de prioridade intermédia e fila de alta prioridade. A fila de baixa prioridade aloja a largura de banda requerida pelo fluxo BE. A fila de prioridade intermédia aloja pedidos de largura de banda enviados pelas ligações rtPS e nrtPS. Estes pedidos das ligações de rtPS e nrtPS podem migrar para a fila de alta prioridade para garantir que os seus requisitos de QoS sejam satisfeitos. Para além dos pedidos vindos das migrações feitas da fila intermédia, a fila de alta prioridade armazena também concessões periódicas e tem a oportunidade de fazer pedidos *unicast* que devem ser geridos na próxima trama. A Figura 62 ilustra a arquitectura do mecanismo de gestão proposto.

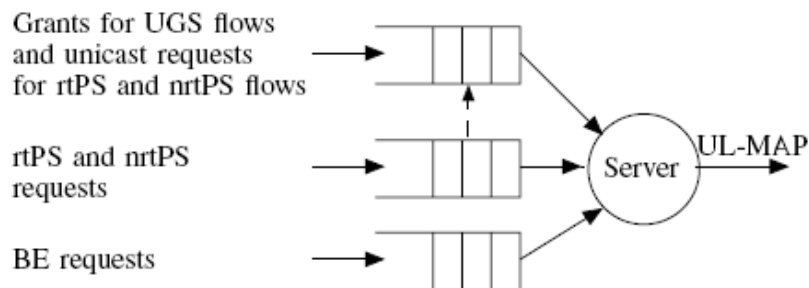


Figura 62 - Arquitectura do mecanismo de gestão proposto [39]

Para garantir a largura de banda para o tráfego UGS, concessões de dados são periodicamente introduzidos nas filas de alta prioridade. O intervalo entre concessões de tráfego UGS é informado pelas SSs no tempo de estabelecimento da ligação. Concessões de pedidos *unicast* também são introduzidas nas filas de alta prioridade. Intervalos entre oportunidades de pedidos para fluxos rtPS e nrtPS são definidos pela BS. Para garantir o parâmetro de máximo atraso requerido, a BS atribui um *deadline* a cada pedido de largura de banda por parte do fluxo rtPS na fila de prioridade intermédia. Cada vez que o mecanismo de gestão é executado, os pedidos que estão com um *deadline* a expirar na segunda trama seguinte migrarão para a fila de alta prioridade. Para calcular o *deadline* para a migração é necessário saber o tempo de chegada dos pacotes nas filas das SSs. Como a BS não tem acesso a essa informação, o pior caso é considerado, correspondendo à chegada na fila imediatamente após a ligação ter enviado o último pedido de largura de banda. Assim, o *deadline* de um pedido deve ser igual à soma do tempo de chegada do último pedido enviado pela ligação e o máximo atraso requerido. O mecanismo de gestão garante a mínima largura de banda requerida por ambas as ligações rtPS e nrtPS sobre uma janela de uma certa duração. Cada vez que o mecanismo de gestão é executado, este calcula um valor de prioridade ao pedido na fila intermédia considerando o requisito mínimo de largura de banda por ligação, a quantidade de pedidos ainda por atender (em bytes) e a quantidade de largura de banda recebida na janela corrente. Valores de baixa prioridade são atribuídos a pedidos enviados pelas ligações em que já receberam a mínima largura de banda requerida na janela corrente [7].

5.3.4. IMPLEMENTAÇÃO

Para a instalação deste módulo no ns-2 é necessário executar os seguintes passos. Depois de descompactada a pasta do módulo, esta apresenta 3 subpastas (Mac, Common e Tcl) que contêm ficheiros que deverão ser introduzidos nas pastas respectivas do ns-2, acrescentando ou substituindo assim os ficheiros já existentes. Depois de efectuar este passo é necessário alterar o ficheiro *Makefile* do ns. Este processo permite que os novos ficheiros introduzidos e os que foram substituídos sejam compilados [40].

A versão do ns-2 usada para o estudo dos módulos e para a sua simulação é a versão 2.32. Sendo este módulo criado para a versão 2.28 foi necessário efectuar algumas alterações na estrutura de certos ficheiros no ns-2.32, tal como na implementação do módulo NDSL, para que o módulo entrasse em funcionamento. Neste ponto foram encontrados alguns

obstáculos devido à falta de informação disponível na internet sobre este módulo, já que este foi disponibilizado muito recentemente. Depois do módulo estar funcional foi necessário efectuar uma investigação dos resultados obtidos em cada simulação de forma a serem produzidas *scripts*, capazes de monitorar correctamente o tráfego da ligação.

5.3.5. EXEMPLO DE UM CENÁRIO DE SIMULAÇÃO COM O MÓDULO UNICAMP

Em seguida serão apresentados alguns excertos de código em linguagem Tcl para a construção de cenários de simulação neste módulo.

Tal como nos dois módulos anteriores o início da geração do cenário é feita através da configuração das opções de simulação. Das opções de configuração possíveis destaca-se a opção de configuração do mac para que o ns-2 seja capaz trabalhar com este módulo

```
set opt(mac) Mac/802_16BS
```

Uma das principais diferenças deste módulo para o módulo implementado pela NDSL reside no facto dos parâmetros de QoS poderem ser alterados a partir do ficheiro Tcl, facilitando este processo. No caso do NDSL é necessário efectuar as alterações desses parâmetros em linguagem C++ provocando a recompilação do módulo para que as alterações entrem em vigor.

```
# Parâmetros de QoS
#UGS
set ugs_grant_interval 0.020
set ugs_grant_size 66

#rtPS
set rtps_grant_interval 0.020
set rtps_delay 0.100
set rtps_aggreq_interval 5
set rtps_minBW 200000

#ertPS
set ertps_grant_interval 0.020
set ertps_grant_size 200

#nrtPS
set nrtps_grant_interval 0.050
set nrtps_minBW 200000
set nrtps_aggreq_interval 10

#BE
set be_grant_interval 2.0
set be_aggreq_interval 10
```

Este exemplo demonstra a introdução de parâmetros de QoS para cada classe de tráfego. Os parâmetros aqui apresentados indicam as limitações necessárias para cada tipo de tráfego funcionar correctamente. Todos os valores introduzidos são em segundos, em *bytes* ou em número de pedidos.

A topologia utilizada para a explicação da geração de cenários no módulo UNICAMP é idêntica à utilizada no módulo criado pela NIST, onde o nó referente à BS está rodeado por *n* nós móveis num ambiente wireless. Esta BS está também ligada através de um ambiente por cabo, a um nó *sink* simulando a ligação da BS à internet. Neste caso, tal como no módulo da NIST, foi utilizado um roteamento hierárquico.

Depois de criado o nó da BS e o nó *Sink* são efectuadas as configurações dos canais e das ligações.

Existem cinco tipos de ligações possíveis, cada uma referindo-se a cada tipo de classes de QoS. O extracto seguinte demonstra a configuração para uma ligação UGS.

```
set ugs_init 0
set ugs_end $opt(n-ugs)

# Configuração da ligação UGS
for {set j $ugs_init} {$j < $ugs_end} {incr j} {

  $node_($j) configure-upflows $node_($j) "0 0 $W
  29 4 0 0 0 0 $ugs_grant_size $ugs_grant_interval
  10000 0 0 0 0 0 0"
  $node_($j) configure-upflows $node_($j) "1 4 $W
  54 4 0 1 1000 1 1500 0 10000 0 0 100 0 0 0"
  $node_($j) configure-downflows $node_($j) "1 4
  $BS 54 4 0 0 10000 0 0"
  $node_($j) startsim
}
```

A configuração de ligação é efectuada através da actualização de inúmeros parâmetros.

Depois da configuração de cada ligação é realizada a configuração de cada tipo de fluxo de tráfego, como demonstra o seguinte exemplo para o caso da classe UGS.

```
# Configuração do tráfego uplink UGS
set ugs_ul_sink [new Agent/LossMonitor]
$ns_ attach-agent $W $ugs_ul_sink

for {set j $ugs_init} {$j < $ugs_end} {incr j} {
  set udp_ul_($j) [new Agent/UDP]
  $udp_ul_($j) set packetSize_ 66
}
```

```

$udp_ul_($j) set class_ 0
$ns_ attach-agent $node_($j) $udp_ul_($j)
$ns_ connect $udp_ul_($j) $ugs_ul_sink
set voice_ul_($j) [new
Application/Traffic/Exponential]
$voice_ul_($j) set packetSize_ 66
$voice_ul_($j) set rate_ 26.4Kb
$voice_ul_($j) set burst_time_ 1.2
$voice_ul_($j) set idle_time_ 1.8
$voice_ul_($j) attach-agent $udp_ul_($j)
set rng_ul_($j) [new RNG]
$rng_ul_($j) seed predef [expr $base_seed+$j]
$voice_ul_($j) use-rng $rng_ul_($j)
}

```

Este procedimento é similar aos explicados nos módulos anteriores bem como no capítulo 3 deste trabalho referente à ferramenta de simulação ns-2. Pode no entanto salientar-se a configuração de aplicações de voz em cada uma das ligações.

5.4. FUNCIONALIDADES DOS MÓDULOS

Durante todo o processo de investigação, simulação e análise de resultados foi-se deparando com vantagens e inconvenientes nos três módulos. Este subcapítulo tem como objectivo relatar, de uma forma sumária, os principais obstáculos encontrados fazendo a distinção entre ambos os módulos.

A Tabela 9 apresenta a comparação entre os três módulos:

Tabela 9 - Comparação dos módulos NDSL, UNICAMP e NIST

	NDSL	UNICAMP	NIST
Facilidade de instalação para a versão 2.32 do ns	Complexo	Muito Complexo	Simples
Complexidade do código C++	Muito Complexo	Complexo	Simples
Complexidade do código Tcl	Simples	Complexo	Simples
Bugs encontrados	Muitos	Poucos	Poucos
Mecanismos de suporte de QoS	Existente	Existente	Sem escalonador
Mobilidade	Inexistente	Inexistente	Existente
Manual de utilização	Inexistente	Inexistente	Existente
Facilidade de criação de cenários	Complexo	Complexo	Complexo
Análise dos resultados	Complexo	Complexo	Simples

Se restringirmos a comparação dos módulos pelos mecanismos de suporte de QoS pode-se afirmar que os módulos da NDSL e da UNICAMP são os indicados para realizar os estudos necessários.

Em relação aos restantes factores referidos na tabela pode-se afirmar que o módulo da NIST apresenta um *interface* com o utilizador muito mais atractivo. Este módulo é capaz de efectuar simulações com mobilidade das SSs (MSs) fazendo *handover* entre BSs.

O código realizado pelos responsáveis do módulo NDSL é bastante complexo obrigando o utilizador a fazer alterações de variáveis em linguagem C++ em vez de serem feitas em código TCL [41][42]. Estas alterações em C++ obrigam a que o ns-2 seja recompilado levando a que este processo seja muito moroso. Os códigos dos módulos implementados pela NIST e pela UNICAMP estão bastante melhor estruturado não demonstrando erros de programação como no módulo da NDSL.

Em relação a manuais de utilização os três apresentam muito pouca informação disponível. O módulo da NDSL apenas se faz acompanhar de um artigo publicado [33] onde os autores demonstram de uma forma geral a implementação do módulo, não proporcionando ao utilizador nenhum manual capaz de o auxiliar fornecendo-lhe a informação necessária à

utilização do módulo. O módulo da UNICAMP é acompanhado por dois artigos [37][39], um referente à construção do módulo e outro referente ao método de escalonamento produzido. Este módulo, tal como o da NDSL, também não vem acompanhado de um manual para que o utilizador aprenda a trabalhar com este módulo. O módulo criado pela NIST está acompanhado por um manual [36]. Apesar de ser uma melhoria comparando com os outros dois módulos, este manual não é completo faltando também muita informação necessária para ser executada uma boa utilização do mesmo. Assim, para utilizadores recém-chegados ao mundo da simulação em ns-2, a exploração destes módulos apresenta um desafio de enorme dificuldade.

No que diz respeito à produção dos cenários de simulação todos os módulos apresentam soluções diferentes, sendo necessária uma boa compreensão do funcionamento de cada módulo para atingir a sua correcta manipulação.

Por último convém referir que na análise de resultados o módulo da NIST vem provido com ferramentas estatísticas, bastando a configuração em TCL para se poder obter os resultados obtidos. No caso da NDSL e da UNICAMP é necessário usar *scripts* de auxílio para efectuar o mesmo. Estes *scripts* foram realizados na linguagem de programação awk.

5.5. RESUMO

O quinto capítulo apresenta os três módulos utilizados para simulação de redes IEEE 802.16 usando o ns-2. Cada um destes módulos integra funções da camada MAC da tecnologia WIMAX seguindo diferentes abordagens para modelização da camada física.

Este capítulo divide-se em três partes onde, em cada uma, é descrito cada módulo. Cada subsecção é composta pela introdução e descrição do módulo, os seus mecanismos de gestão de QoS, a sua implementação e ainda a exposição de um exemplo de um cenário de simulação definido para o módulo em questão.

Um dos módulos considerados, desenvolvido pelo *Networks & Distributed Systems Laboratory* (NDSL) implementa o modo TDD (*Time Division Duplexing*) e a topologia PMP (*Point-to-Multipoint*) utilizando a implementação do canal sem fio já fornecida pelo ns-2. Este modelo é capaz de implementar os cinco tipos de serviço definidos pela norma 802.16.

Outro módulo estudado, implementado pelo *National Institute of Standards and Technology* (NIST), fornece, entre outras funcionalidades, a camada física OFDM com modulação configurável, TDD, topologia ponto-a-multiponto (PMP) e fragmentação de pacotes. No entanto, este modelo não implementa os mecanismos para suporte de QoS na camada MAC, isto é, os diferentes tipos de serviço e mecanismos de escalonamento com QoS.

O último módulo a ser disponibilizado ao público foi o desenvolvido pela UNICAMP, Universidade Estadual de Campinas, Brasil. Tal como no módulo da NDSL este fornece um mecanismo de escalonamento para o tráfego *uplink* contemplando cada uma das cinco classes previstas pela norma. Este módulo segue as especificações da norma IEEE 802.16 para a topologia PMP e o modo de acesso utilizado é o TDD.

Por fim é introduzida uma subsecção onde se efectua a comparação do funcionamento dos módulos, relatando as suas vantagens e principais inconvenientes encontrados ao longo do processo de investigação, simulação e análise de resultados.

6. SIMULAÇÃO & ANÁLISE DE RESULTADOS

Este capítulo reporta os testes realizados com os módulos WIMAX de forma a permitir uma análise das características de cada um através de simulações na ferramenta ns-2. Com este propósito, os vários módulos foram ajustados de modo a poderem ser executados com a versão 2.32 do ns-2.

As simulações efectuadas têm como objectivo principal comparar os módulos implementados pela NDSL e pela UNICAMP, detentores de mecanismos de gestão das classes de QoS para o tráfego *uplink*, verificando se estes são capazes de assegurar as especificações da norma 802.16 conforme é indicado nos documentos que os acompanham.

O módulo criado pela NIST, sem um mecanismo de gestão de classes de QoS, é incapaz de proporcionar a qualidade de serviço necessária para assegurar estas mesmas especificações, sendo posto de parte nesta avaliação. Foram testados os módulos da NDSL e UNICAMP avaliando o seu comportamento de uma forma comparativa a fim de aferir qual apresenta uma melhor estratégia de escalonamento de cada classe de QoS.

Foram elaboradas várias simulações, em diferentes cenários, de forma a demonstrar o pretendido. Em particular, para topologias ponto-a-multiponto consideram-se cinco tipos de tráfego: voz, VoIP, vídeo, FTP e WEB *browsing*, que estão associados respectivamente às cinco classes de QoS definidas pela norma 802.16 – UGS, ertPS, rtPS, nrtPS, e BE.

Foi testado também se a BS de cada módulo é capaz de assegurar, de uma forma justa, a largura de banda partilhada por diversos fluxos de tráfego competidores entre si, pertencendo ao mesmo grau de serviço (classe de serviço BE). Nesta experiência, para além dos módulos da NDSL e da UNICAMP, foi também possível testar o módulo implementado pela NIST já que este trata todos os tráfegos de igual modo, de uma forma BE (*Best Effort*).

Todos os cenários de simulação consistem numa BS com várias SSs uniformemente distribuídas ao redor dela. Os cenários não foram escolhidos com a intenção de simular redes operacionais reais, mas sim com o intuito de analisar de uma forma teórica, o comportamento dos módulos, tanto ao nível dos mecanismos de escalonamento de QoS como na alocação da largura de banda entre diversas ligações concorrentes. De forma a prevenir que o escalonamento de pacotes nas SSs não interfira com a evolução do mecanismo de escalonamento implementado na BS, nos testes realizados assume-se que cada SS apenas possui um fluxo de tráfego.

As classes de serviço de cada um dos módulos são definidas de acordo com a visão dos autores, no que respeita às especificações de cada tipo de tráfego. Nos testes realizados procurou-se utilizar os mesmos parâmetros para ambos os módulos de forma a torná-los comparáveis.

A Figura 63 mostra, através da ferramenta nam, um exemplo possível de um cenário de simulação. É retratado um cenário onde o nó 0 corresponde à BS estando rodeada por 5 SSs, onde cada SS possui um tipo de tráfego diferente.

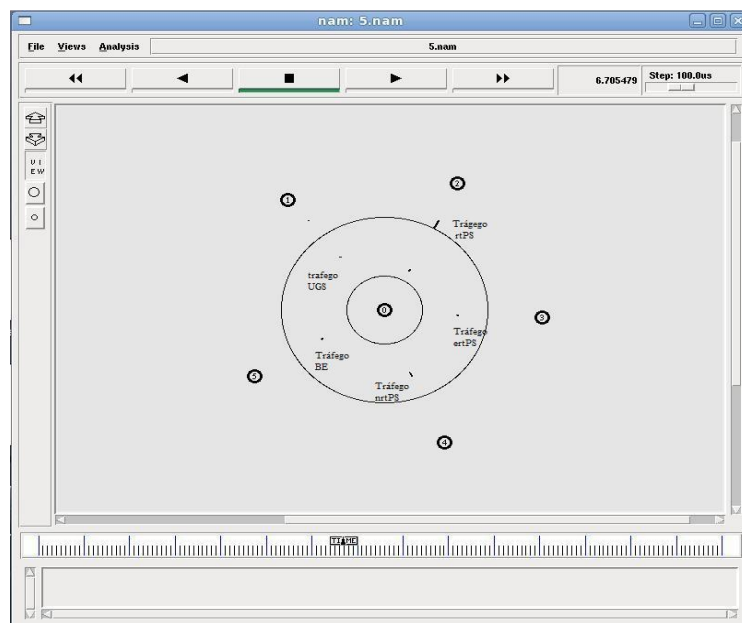


Figura 63 - Cenário 1 BS e 5 SS com diferentes tráfegos a serem gerados

O comportamento de cada módulo foi avaliado com base na análise da taxa de transferência efectiva (*throughput*) que cada SS dispunha para transmitir o seu tráfego de *uplink*, bem como, da análise dos atrasos existentes nas ligações de maior prioridade.

Para analisar os resultados obtidos através das simulações foi necessário desenvolver *scripts* em linguagem de programação awk que permitisse extrair os valores dos ficheiros *output* das várias simulações.

No decorrer do estudo realizado com cada módulo, foi também objectivo do trabalho, analisar o seu modo de funcionamento, aferindo as suas principais vantagens e os seus inconvenientes.

Os módulos testados foram desenvolvidos com base em versões diferentes do ns. Na secção 6.2 procura-se mostrar que as adaptações introduzidas em cada módulo não alteram o seu funcionamento de modo expressivo, uma vez que se mostra que os resultados encontrados, após adaptação deste trabalho, encontram-se de acordo com os resultados apresentados pelos autores dos módulos.

6.1. MEDIÇÕES UTILIZADAS

6.1.1. TAXA DE TRANSFERÊNCIA EFECTIVA

A taxa de transferência efectiva (*throughput*) pode ser definida como a quantidade de dados transferidos entre dois pontos ou a quantidade de dados processados num determinado espaço de tempo. Numa rede de dados, este conceito é empregue quando se refere à quantidade de dados recebidos no destino. Este valor poderá ser diferente da quantidade de dados enviados pela origem devido a perdas de pacotes, duplicação dos mesmos ou atrasos do sistema. As unidades básicas são o kbps, Mbps ou o Gbps. É importante fazer a distinção entre taxa de transferência efectiva e largura de banda. A largura de banda representa a maior capacidade que pode ser obtida através de uma transferência descorando o facto de os dados recebidos serem os relevantes. A taxa de transferência efectiva ou débito representa a velocidade a que um computador consegue processar dados combinado com a eficiência do *software* que actua sobre ele. No caso deste trabalho de investigação, a taxa de transferência efectiva foi calculada através da soma de todos os pacotes recebidos (em bits) na camada de aplicação correspondente ao fluxo de tráfego em questão por unidade de tempo.

6.1.2. ATRASO MÉDIO

Neste trabalho considera-se o atraso como sendo o intervalo de tempo que um pacote demora a atravessar uma ligação numa rede. Este tempo de atraso foi medido como o intervalo de tempo desde a emissão de um pacote gerado por uma SS até à sua recepção na BS. Para cada tipo de tráfego foram medidos os valores do tempo que era despendido no processamento de cada pacote. No final de cada simulação foi obtido o valor médio dos atrasos adquiridos. Como se pode verificar na secção 4.7 deste trabalho, o atraso médio foi calculado como a média da soma dos intervalos de tempo que um pacote despendia a percorrer uma ligação desde a sua geração, no nível da aplicação de uma SS, até à sua chegada no mesmo nível na BS. Para que a comparação entre os módulos do WIMAX fosse rigorosa definiu-se que as SSs estão todas à mesma distância da BS.

6.2. ANÁLISE DOS VALORES PUBLICADOS

Este subcapítulo tem como objectivo testar se os resultados obtidos nas simulações dos sistemas ajustados à versão 2.32 do ns correspondem aos valores encontrados pelos autores na apresentação dos seus módulos. Para atingir este fim, foram efectuadas simulações de cenários, criados pelos autores, e comparados os resultados obtidos com os resultados apresentados nos artigos científicos ou manuais que acompanham cada módulo.

6.2.1. MÓDULO NDSL

No artigo [33] que acompanha o módulo criado pela NDSL está presente uma simulação que relaciona o atraso médio dos pacotes que percorrem uma ligação com o aumento do número de SSs presentes na rede para o tráfego UGS. O cenário simulado consiste numa BS que está a servir 10 SSs. As SSs estão distribuídas aleatoriamente em redor da BS. O módulo faz-se acompanhar por este cenário de simulação, sendo possível efectuar a sua replicação de forma a testa-la na versão 2.32 do ns. A Figura 64 ilustra os resultados obtidos nas simulações efectuadas pelos autores do módulo.

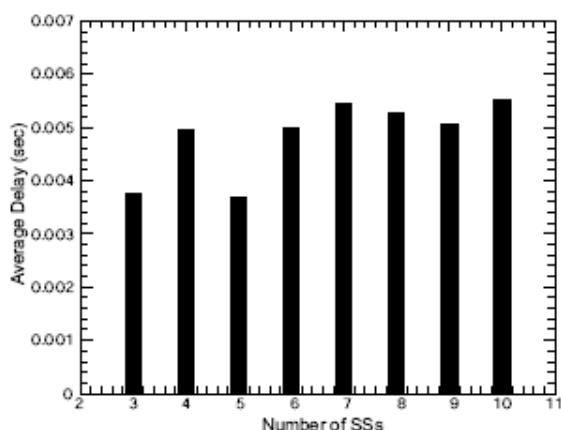


Figura 64 - Atraso médio em função do nº de SSs (NDSL)

É possível verificar que o atraso médio apresentado varia entre os 3 e os 5 ms, evidenciando um ligeiro aumento com o aumento do número de SSs presente na rede.

A Figura 65 mostra os resultados obtidos com a utilização do mesmo cenário na versão 2.32 do ns.

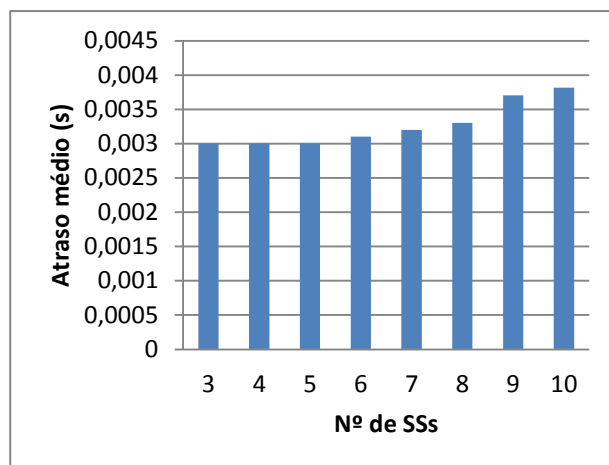


Figura 65 - Atraso médio em função do nº de SSs b)

Como é possível constatar, os valores encontrados são muito similares aos apresentados no artigo que acompanha o módulo NDSL. Nestas simulações verificaram-se valores um pouco inferiores aos encontrados no artigo. É natural que esta diferença exista já que os autores do módulo não indicam qual o critério usado no cálculo do atraso médio encontrado podendo haver diferenças entre as *scripts* utilizadas.

6.2.2. MÓDULO UNICAMP

Os autores do módulo da UNICAMP produziram um artigo científico [39] que relata várias experiências que demonstram o funcionamento do escalonar de QoS. Foi escolhida uma das experiências desse artigo e reproduzida com o módulo ajustado para a versão 2.32 do ns de forma a verificar se os resultados obtidos correspondem aos apresentados no artigo.

O cenário descrito tem como objectivo analisar se o aumento de ligações UGS prejudica o atraso em ligações de alta prioridade como o rtPS e o UGS. A estrutura da rede é composta por uma BS e 81 SSs. Nesta experiência existem 6 ligações rtPS, 20 ligações nrtPS, 20 ligações BE, e o número de ligações UGS activas varia de 15 a 35.

A Figura 66 ilustra os resultados obtidos pelos criadores do módulo.

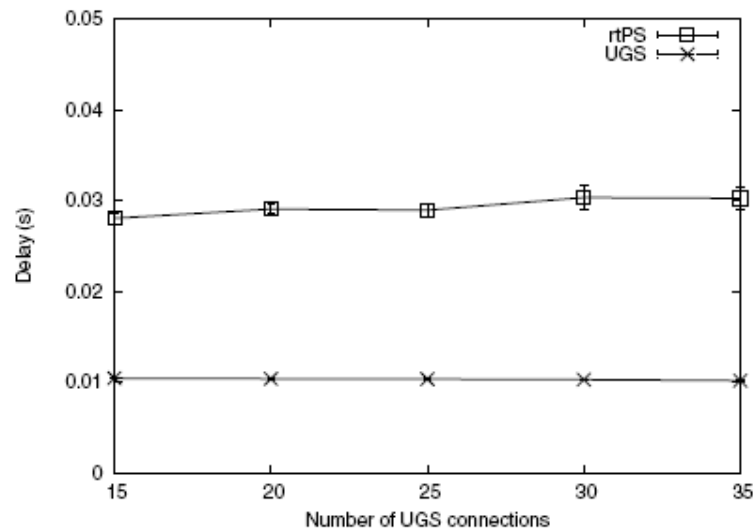


Figura 66 - Atraso das ligações rtPS e UGS (UNICAMP)

Ao analisar o gráfico verifica-se que o atraso apresentado para as ligações UGS não apresenta variação significativa, pelo que estas ligações não são afectadas pelo aumento carga na rede. O atraso para as ligações rtPS apresenta uma ligeira oscilação à medida que a carga vai aumentando. Contudo, estes atrasos são consideravelmente baixos comparados com os valores requeridos.

A Figura 67 apresenta os resultados obtidos na tentativa de replicação da experiencia anteriormente apresentada pelos autores do módulo.

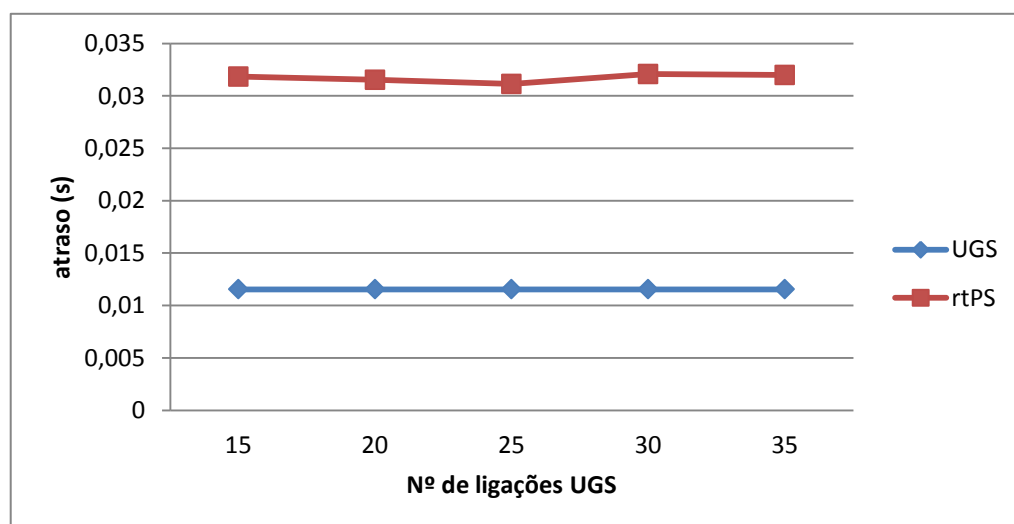


Figura 67 - Atraso das ligações rtPS e UGS b)

Analisando o gráfico é possível constatar que os valores para as ligações UGS e rtPS apresentam um nível aproximadamente constante de atraso, sendo ligeiramente superiores

ao apresentado no artigo científico. Esta pequena diferença encontrada deve-se, provavelmente, ao facto das scripts desenvolvidas para o cálculo do atraso serem diferentes das utilizadas pelos autores do módulo. A não especificação do tempo de simulação no artigo científico, juntamente com o facto deste módulo ser implementado numa versão do ns-2 diferente do testado no artigo, são, também, possíveis justificações para a ligeira diferença de resultados encontrados. Assim é patente que as alterações efectuadas no módulo ao longo deste trabalho não alteram de forma significativa o seu funcionamento.

6.2.3. MÓDULO NIST

A documentação que acompanha o módulo da NIST [36] não apresenta nenhum resultado de testes que os autores tenham realizado. Porém, nos exemplos os autores indicam que o número de pacotes que cada SS envia e que cada SS recebe é 100, não havendo perdas de pacotes por colisão na requisição de largura de banda. Para comprovar o facto foram efectuadas simulações com o intuito de atingir o resultado indicado nos exemplos. Utilizando o exemplo test-be.tcl que já vinha com o módulo, foi possível obter resultados idênticos aos esperados. Nestas simulações verifica-se que o aumento do número de nós leva a um aumento da quantidade de pacotes a serem enviados numa forma proporcional. Cada SS envia 100 pacotes que vão ser recebidos pela BS. Foi possível verificar que o nº de pacotes enviados é idêntico ao nº de pacotes recebidos significando que não houve qualquer perda por colisão na requisição de largura de banda.

Depois dos 3 módulos estarem a comportar-se de acordo com os resultados publicados pelos autores dos mesmos foi, então, possível passar para a simulação dos cenários demonstrados a seguir.

6.3. 1º CENÁRIO DE SIMULAÇÃO – ESTUDO DAS DIFERENTES CLASSES DE SERVIÇO

O primeiro cenário apresenta um caso de múltiplos serviços. Neste caso o fornecedor de largura de banda (BS) tem que suportar ligações com diferentes classes de serviço, detentoras de diferentes características.

O objectivo deste cenário é testar se o escalonador de cada módulo, implementado na BS, é capaz de ter em consideração os diferentes tipos de classes de QoS, alocando os *slots* de acordo com as especificações requeridas por cada uma dessas classes.

A Tabela 10 representa os diferentes parâmetros necessários para o correcto funcionamento de cada tipo de serviço.

Tabela 10 - Requisitos das classes de QoS

	Taxa de transferência	Atraso
UGS	√ Máxima taxa de transferência igual à mínima taxa de transferência	√ < 20 ms
ertPS	√	√ < 50 ms
rtPS	√	√ < 60 ms
nrtPS	√	
BE		

De acordo com as recomendações [6][7][39][43][44] o serviço UGS requer que a taxa máxima de transferência efectiva seja idêntica à taxa mínima de transferência efectiva, tornado o seu débito constante. O atraso admitido por este serviço deverá ser inferior a 20ms. Para a classe ertPS o atraso máximo admitido deverá ser inferior a 50ms. A norma indica também que é necessário satisfazer a taxa mínima de transferência nesta mesma classe. A classe rtPS apresenta 60ms como o limite máximo de atraso e que cada ligação deverá satisfazer uma taxa mínima de transferência efectiva que varia de acordo com a taxa de transmissão de vídeo. O serviço nrtPS apresenta, como requisito uma largura de banda superior a 5kbs. No caso da classe BE, a norma não propõe nenhuma especificação.

O tempo de simulação para cada análise do primeiro cenário é de 200s. Para cada análise foram realizadas 5 simulações e considerada a média dos valores obtidos.

Na estrutura de rede para este cenário de simulação está presente uma BS rodeada por 5SSs. Cada uma das SSs apresenta apenas um tipo de tráfego. A estrutura da rede também inclui um nó ligado por cabo à BS, ao qual todas as SSs enviam os seus dados. Esta ligação por cabo entre o nó, que simula a rede núcleo, e a BS, possui uma largura de banda e um atraso de 1Gbps e 2ms, respectivamente. Foi configurada a largura de banda para esta ligação para que o seu valor fosse suficientemente elevado comparativamente com a largura de banda presente na rede WIMAX. Com isto, pretendeu-se minimizar os

“engarrafamentos” que poderiam ser encontrados se a largura de banda no cabo fosse inferior à utilizada pela rede WIMAX. Assim, foi possível testar se o escalonador na BS é capaz de alocar o número de *slots* suficientes para todas as SSs alocando os recursos necessários para o bom funcionamento de cada uma das classes de QoS.

Neste cenário apenas serão efectuados estudos nos módulos possuidores de mecanismos de escalonamento de classes de QoS, o módulo NDSL e o módulo UNICAMP. Cada módulo apresenta diferentes formas de configuração dos parâmetros de simulação. Para além deste factor convém mencionar que certos parâmetros são passíveis de serem alterados num módulo e não no outro. O módulo implementado pela NDSL não permite, por exemplo, que o utilizador configure certos requisitos de QoS, tais como atrasos máximos ou largura de banda mínima para as diferentes classes de serviço. Tendo em consideração estas diferenças, foi feita a tentativa de equivaler os tráfegos de cada módulo de forma a ser possível efectuar uma comparação justa.

Em ambos os módulos, a classe UGS apresenta, como parâmetros de configuração, um intervalo entre *bursts* de dados de 20ms, e gera tráfego E1/T1 a uma taxa de 65kbps possuindo um tamanho fixo dos pacotes. À classe ertPS é lhe concedida o mesmo intervalo entre *bursts* de dados gerando tráfego VoIP a uma taxa de 65kbps. No módulo implementado pela UNICAMP é possível configurar o atraso máximo admissível, que neste caso foi de 50ms. A classe rtPS gera um tráfego MPEG a uma taxa de 1400kbps. Mais uma vez, no módulo UNICAMP foi possível configurar o atraso máximo admissível, 100ms e a largura mínima aconselhada para este tipo de tráfego, 100 kbps. A classe nrtPS possui uma aplicação FTP que gera tráfego a uma taxa de 100kbps. Por fim, a classe de menor prioridade, BE, apresenta uma aplicação com um padrão de tráfego do tipo *Web Browsing* gerado a uma taxa de 50kbps.

No decorrer da simulação serão analisadas as taxas de transferência efectiva de cada SS bem como os atrasos referentes às classes de maior prioridade.

6.3.1. ANÁLISE I – ESTUDO DO COMPORTAMENTO DAS 5 CLASSES DE SERVIÇO

A Figura 68 representa a topologia de rede para a primeira análise efectuada no primeiro cenário. A rede é composta por uma BS rodeada por 5 SSs, cada uma com um único tipo de tráfego. A BS apresenta também, uma ligação por cabo a um nó de destino.

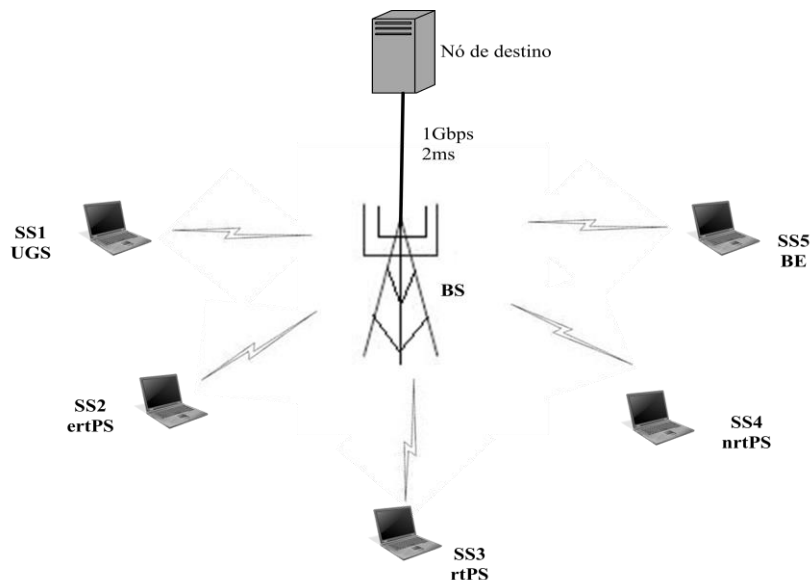


Figura 68 - Estrutura de rede – 1º cenário (análise I)

Cada uma das cinco SSs possui uma aplicação que gera padrões de tráfego correspondentes aos diferentes tipos de serviços. A Tabela 11 apresenta a correspondência entre cada classe de QoS com cada aplicação.

Tabela 11 - Aplicações das SSs

	SS1	SS2	SS3	SS4	SS5
Aplicação	E1/T1	VoIP	Vídeo	FTP	WEB <i>browsing</i>
Classe de QoS	UGS	ertPS	rtPS	nrtPS	BE

RESULTADOS OBTIDOS – MÓDULO NDSL

A Figura 69 apresenta os resultados medidos para a taxa de transferência de cada SS para o tráfego *uplink* em função do tempo de simulação.

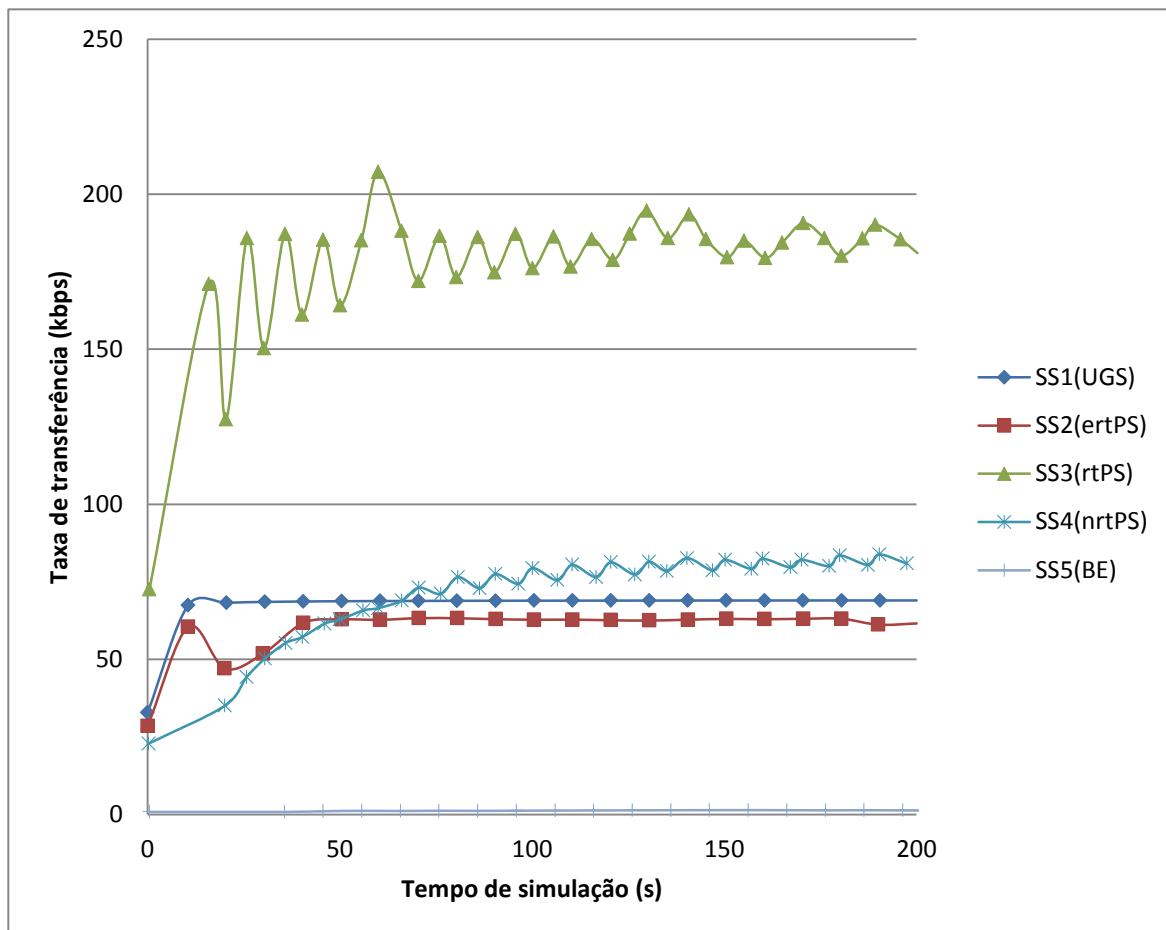


Figura 69 - Taxa de transferência efectiva de 5SSs (NDSL)

Analisando o gráfico verifica-se que o tráfego UGS apresenta uma taxa de transferência efectiva na ordem dos 65 kbps. A variação entre as taxas de transferência mínimas e máximas é de aproximadamente 1kbps. Este valor não é significativo levando a concluir que este módulo, no que se refere à taxa de transferência para a classe UGS, apresenta valores que estão de acordo com as especificações da norma. No início do tempo de simulação este tráfego, tal como os restantes, apresenta níveis inferiores que se justificam pelo tempo perdido pelas SSs no período de *ranging* e de requisição de largura de banda.

O tráfego ertPS, característico de comunicações VoIP, apresenta valores similares aos encontrados pelo serviço UGS tendo um nível de taxa de transferência efectiva ligeiramente inferior, demonstrando que esta classe, apesar de lhe ser submetido a mesma taxa de tráfego que no caso UGS, apresenta um nível de prioridade inferior. É possível analisar, também, uma ligeira oscilação inicial.

A SS3 possui os valores mais elevados, na ordem dos 180kbps. Apresenta variações elevadas em relação aos mínimos e máximos, o que é compreensível para este tipo de comunicações em tempo real. É possível constatar que o valor da taxa de transferência efectiva encontra-se muito abaixo da taxa submetida, demonstrando que este módulo é incapaz de conceder à classe rtPS altas taxas de transferência efectiva, limitando o tráfego desta a classe próxima dos 200kbps. Mesmo assim, este módulo é capaz de proporcionar ao tráfego rtPS uma taxa de transferência superior às restantes, conveniente para se poder efectuar serviços em tempo real.

O tráfego nrtPS apresenta valores acima dos valores mínimos requeridos pela norma (5kbps). É possível constatar que este serviço apresenta valores que seguem a variação do serviço rtPS, já que é o serviço de maior prioridade que apresenta variações mais acentuadas, levando a que o tráfego nrtPS aproveite as fases de menor ocupação da largura de banda das ligações de maior prioridade para atingir os seus máximos.

O serviço BE é o que apresenta menor prioridade entre as cinco classes de QoS disponíveis. Os valores da sua taxa de transferência estão na ordem de 1kbps.

RESULTADOS OBTIDOS – MÓDULO UNICAMP

A Figura 70 ilustra o desenvolvimento da taxa de transferência efectiva em função do tempo para o módulo UNICAMP.

Para melhor visualização dos tráfegos que apresentam valores baixos de taxa de transferência, o gráfico da Figura 70 foi decomposto em mais dois gráficos representados pela Figura 71, onde apresenta os tráfegos UGS e ertPS, e a Figura 72, que apresenta os tráfegos de menor prioridade nrtPS e BE.

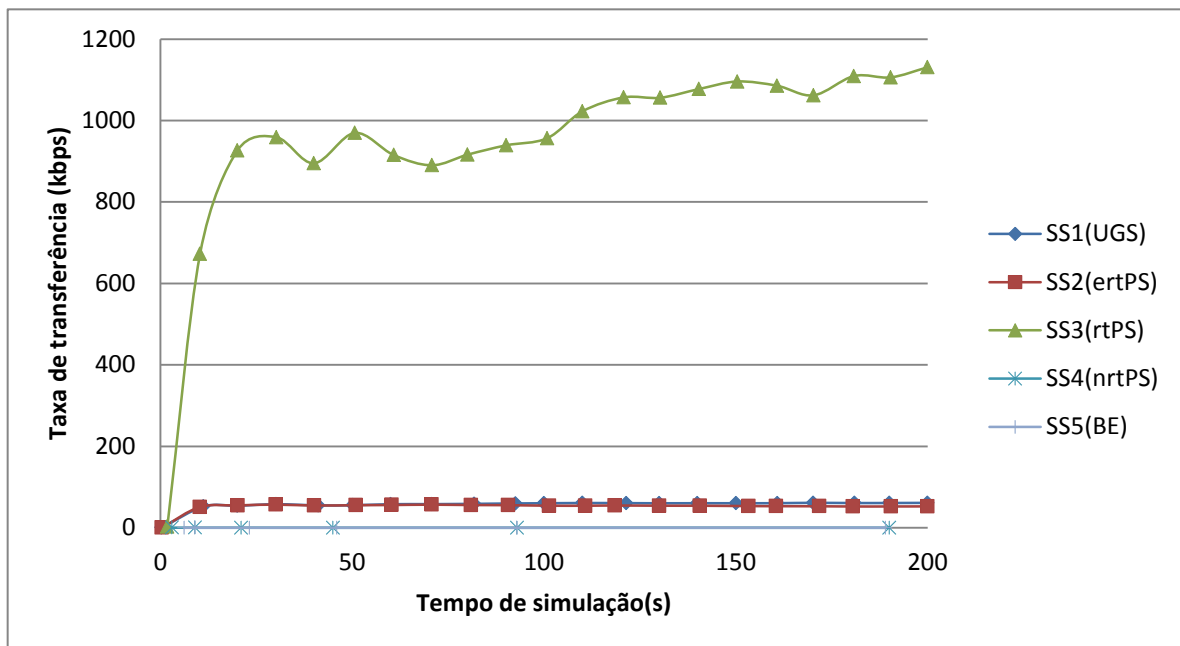


Figura 70 - Taxa de transferência efectiva de 5SSs (UNICAMP)

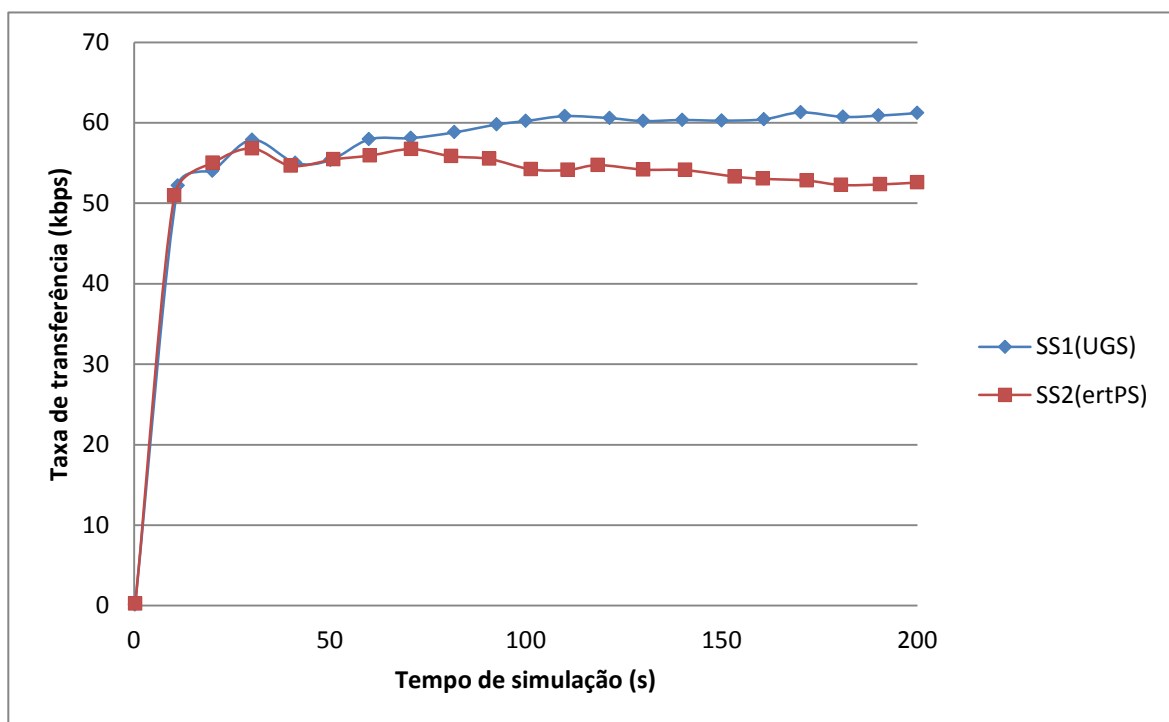


Figura 71 - Taxa de transferência efectiva de 5SSs b) (UNICAMP)

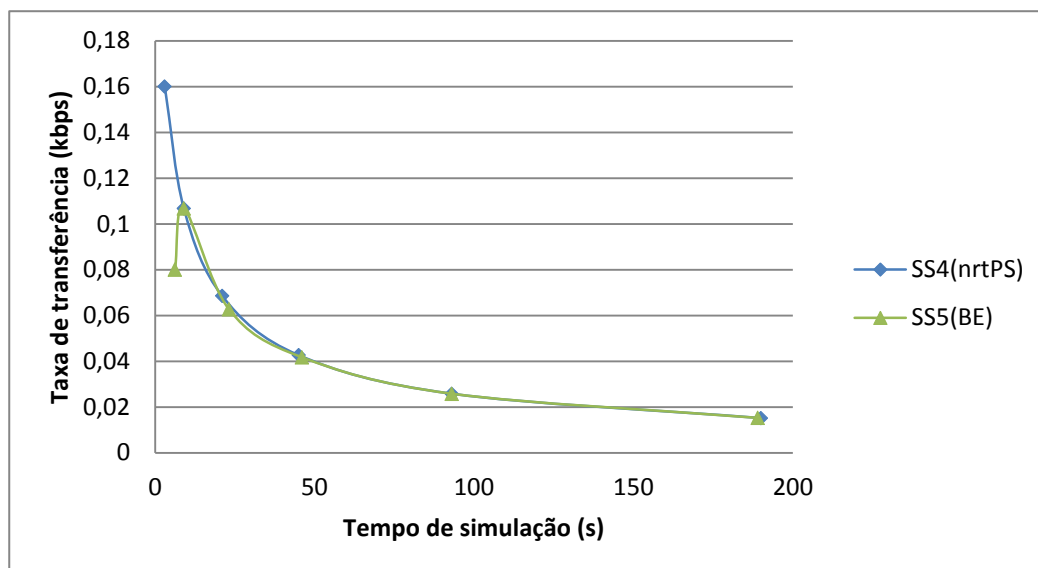


Figura 72 - Taxa de transferência efectiva de 5SSs c) (UNICAMP)

Como é possível verificar a partir da Figura 71 o tráfego UGS, depois de passados os primeiros 100s de simulação apresenta sempre uma taxa de transferência efectiva de 60kbps. Neste módulo é importante constatar que apenas a meio da simulação o sistema encontra a sua estabilidade. A partir deste ponto a variação encontrada entre os valores mínimos e máximos da taxa de transferência é apenas de 700 bps demonstrando que o escalonador da BS é capaz de alocar, sempre, o número de *slots* necessário ao bom funcionamento desta classe de maior prioridade.

A Figura 71 ilustra também o tráfego ertPS. Neste caso, similarmente ao que sucede com o tráfego UGS, só a partir do meio da simulação (100s) é que o tráfego estabiliza. A sua taxa de transferência efectiva ronda os 55kbps. Até ao fim da simulação esta decresce 2kbps.

Como seria de esperar o tráfego rtPS (Figura 70) apresenta os maiores valores de taxa de transferência efectiva, chegando a atingir 1.1 Mbps. Os valores mínimos estão muito acima dos valores requeridos pela norma demonstrando que o escalonador, respeitante à taxa de transferência efectiva, é capaz de assegurar o bom funcionamento da classe rtPS.

A Figura 72 refere-se às SS que possuem aplicações de menor prioridade. Analisando o gráfico, é possível verificar que o módulo implementado pela UNICAMP apresenta um tratamento similar a estas duas classes de tráfego. Estes, quando estão numa rede que é partilhada com vários tipos de tráfego com prioridades superiores, vêem as suas taxas de transferência decrescerem impossibilitando o correcto funcionamento das suas aplicações.

ATRASOS

A Figura 73 demonstra a comparação dos atrasos das classes de maior prioridade entre os módulos implementados pela NDSL e pela UNICAMP

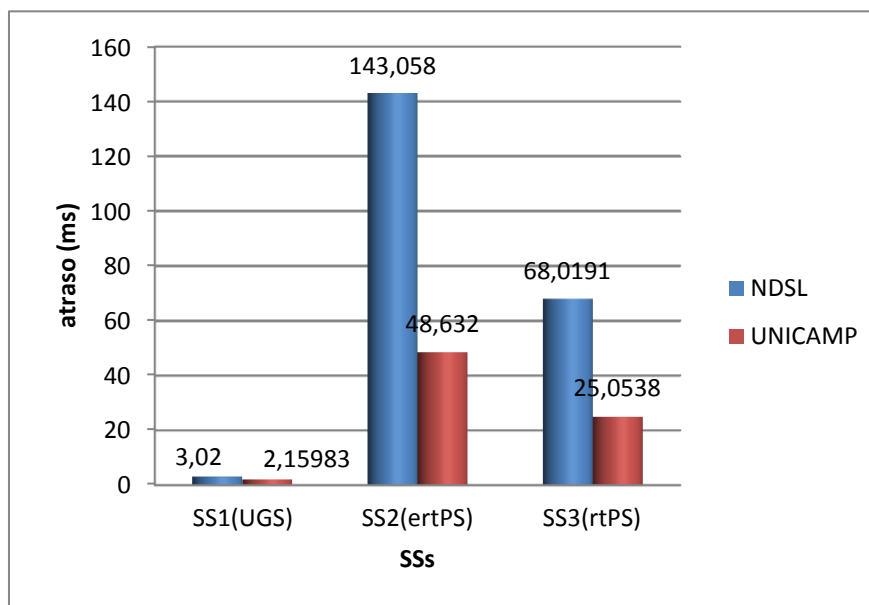


Figura 73 - Atrasos referentes ao 1º cenário, análise I

Relativamente à classe de maior prioridade (UGS), ambos os módulos apresentam valores de atraso médio dentro dos valores requeridos. Apresentam atrasos médios de 3.02ms e 2.16ms nos módulos implementados pela NDSL e pela UNICAMP respectivamente. Estes resultados indicam que os módulos apresentam atrasos médios na classe UGS muito inferiores ao valor máximo exigido pela norma (20ms).

O módulo NDSL apresenta um valor de atraso referente à classe ertPS muito acima do requerido neste tipo de aplicações (50ms) demonstrando que este módulo é incapaz de fornecer à classe ertPS a qualidade exigida para aplicações como o VoIP. Este facto deve-se a que este módulo não permite que o utilizador configure os atrasos máximos para qualquer tipo de serviço. No módulo da UNICAMP o atraso médio é de 48.632ms estando no limiar do exigido para este tipo de aplicações.

Para aplicações que operam na classe rtPS os atrasos deverão ser inferiores a 60ms. O módulo UNICAMP apresenta um valor inferior ao requerido na SS que suporta esta classe exibindo um atraso médio de 25ms. O módulo NDSL foi incapaz de corresponder ao exigido. Neste tipo de tráfego apresenta valores de atraso médio de 68ms.

COMPARAÇÃO DO COMPORTAMENTO DOS MÓDULOS NA 1ª ANÁLISE DO 1º CENÁRIO

Nesta simulação foi possível constatar que ambos os módulos apresentam um bom funcionamento para a classe UGS. Neste caso, a taxa de transferência efectiva foi mantida constante ao longo de todo o processo de simulação. Em relação à classe ertPS, apesar de os valores apresentados da taxa de transferência efectiva estarem de acordo com as especificações (Tabela 10), foi possível identificar que os atrasos encontrados, no módulo NDSL, neste tipo de ligações são superiores aos requeridos. No que diz respeito à classe rtPS os módulos apresentam os valores mais altos de taxa de transferência efectiva, chegando a atingir, para o caso do módulo UNICAMP valores de 1.1Mbps. Relativamente aos atrasos encontrados nestas ligações rtPS é possível verificar que o módulo UNICAMP apresenta valores dentro dos padrões propostos enquanto o módulo NDSL não. Analisando o gráfico (Figura 72) verifica-se que no caso das classes de menor prioridade (nrtPS e BE) o escalonador do módulo UNICAMP é incapaz de salvaguardar os requisitos necessários para o funcionamento destas duas classes. O módulo NDSL, devido à sua política de escalonamento em *rounds (weighted round-robin)*, que atribui no primeiro *round* uma pequena percentagem de banda a cada classe, foi possível verificar que os resultados apresentados foram suficientes para o funcionamento destas duas classes.

6.3.2. ANÁLISE II – ESTUDO DO COMPORTAMENTO DA CLASSE UGS E BE

A topologia de rede nesta análise é composta por 5 SSs onde 4 estabelecem ligações UGS com a BS e uma SS estabelece uma ligação BE. Também neste caso a BS está ligada por cabo a um nó de destino. A Figura 74 apresenta a topologia utilizada na criação da segunda análise.

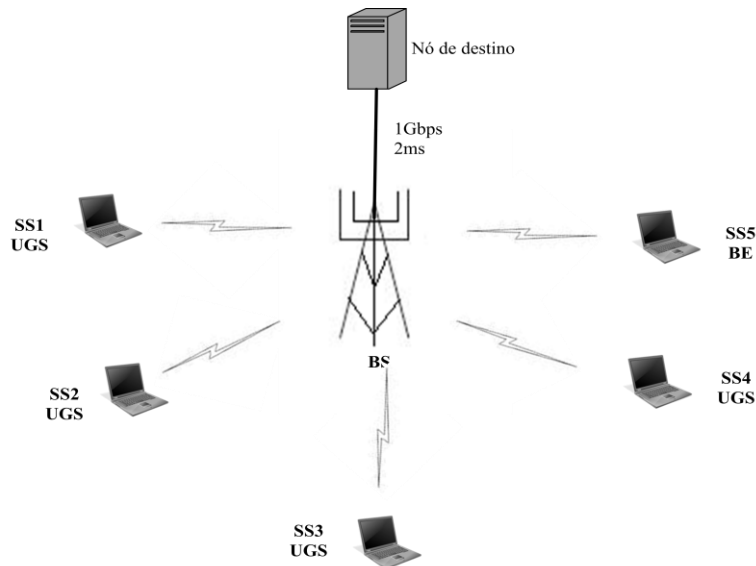


Figura 74 - Estrutura da rede – 1º cenário (análise II)

Com esta análise pretendeu-se verificar se a BS é capaz de alocar o número suficiente de *slots* a cada uma das ligações UGS, verificando-se também se o serviço BE ainda fica com alguns *slots* para transmissão de tráfego no *uplink*. Utilizaram-se as 4 SSs com o tráfego UGS uma vez que este apresenta o maior nível de prioridade e necessita de uma taxa de transferência eficaz sempre constante.

O tempo utilizado para a simulação do terceiro cenário foi de 200ms tendo sido realizadas 5 simulações considerando-se a média dos valores obtidos.

RESULTADOS OBTIDOS – MÓDULO NDSL

A Figura 75 apresenta os resultados referentes à taxa de transferência efectiva do módulo implementado pelo NDSL em função do tempo.

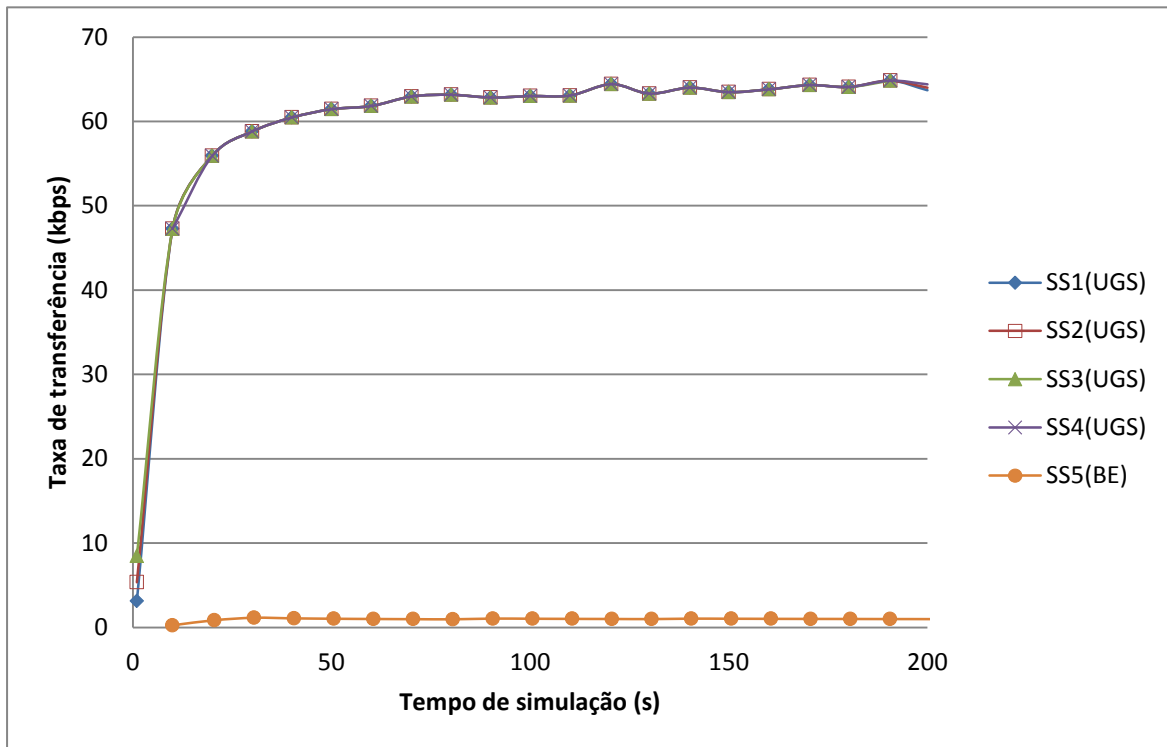


Figura 75 - Taxa de transferência efectiva para 5 SSs (NDSL)

Os resultados deste teste mostraram que, independentemente do número de SSs na rede com o tráfego UGS, este módulo é capaz de efectuar sempre a alocação de *slots* necessária para o bom funcionamento deste tipo de serviço. Verifica-se que a BS atribui a largura de banda ao tráfego UGS de uma forma justa, apresentando, todas as SSs com ligações UGS, o mesmo nível de taxa de transferência efectiva, 62kbps. É importante referir, que nesta simulação com o módulo da NDSL, os pacotes são enviados pelas diferentes SSs no mesmo tempo levando a BS a receber os pacotes praticamente em simultâneo havendo uma diferença mínima, como demonstra o seguinte extracto de um dos *trace files* (subsecção 4.6.1) executados nesta análise.

```
s -t 20.000000000 -Hs 1 (...)
s -t 20.000000000 -Hs 2 (...)
s -t 20.000000000 -Hs 3 (...)
s -t 20.000000000 -Hs 4 (...)

r -t 20.003080702 -Hs 0 (...)
r -t 20.003080703 -Hs 0 (...)
r -t 20.003080705 -Hs 0 (...)
r -t 20.003080708 -Hs 0 (...)
```

Analisando o extracto do trace file verifica-se que, num determinado instante, cada SS (nó 1, 2, 3, 4) envia um pacote UGS, levando a BS (nó 0) a receber cada um com uma diferença de apenas poucos nanossegundos. Este facto leva a que o gráfico da taxa de

transferência efectiva apresente valores idênticos para cada ligação UGS, uma vez que a recepção dos pacotes é feita praticamente em simultâneo originando o mesmo valor de taxa de transferência em cada instante e para cada ligação.

A ligação BE também apresentou um nível constante da taxa de transferência efectiva, atingindo valores na ordem do 1kbps.

RESULTADOS OBTIDOS – MÓDULO UNICAMP

A taxa de transferência efectiva para a análise II, referente ao módulo UNICAMP, é apresentada na Figura 76. Os resultados obtidos foram adquiridos numa simulação com um período de 200s.

Foi necessário efectuar o desdobramento do gráfico seguinte de forma a poder ser analisado o comportamento do tráfego *uplink* efectuado entre a SS5 (BE) e a BS como demonstra a Figura 77.

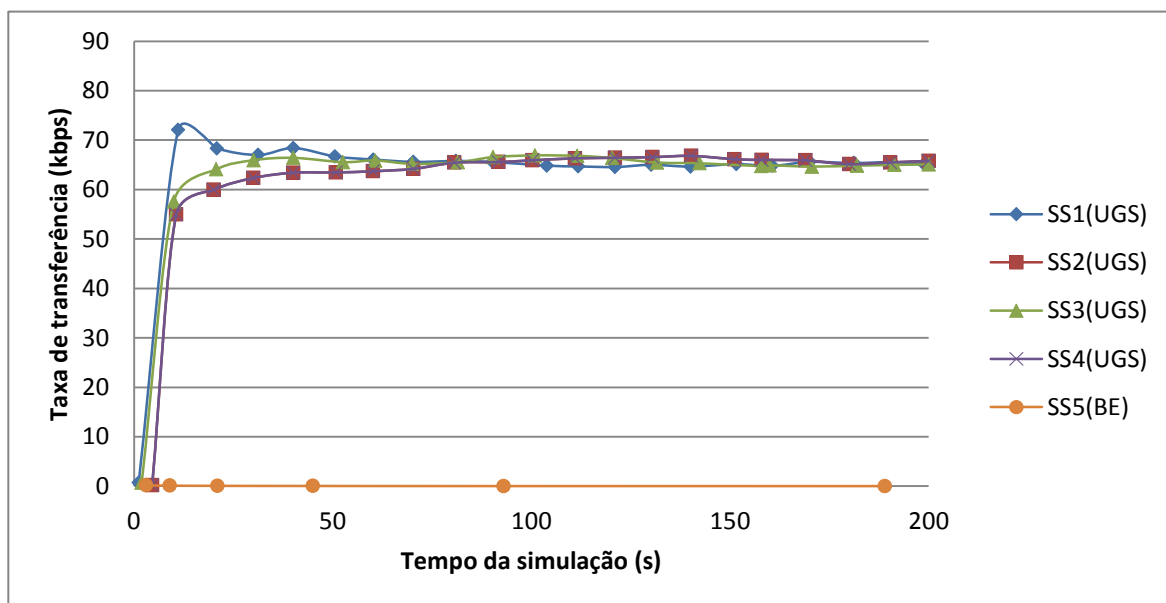


Figura 76 - Taxa de transferência efectiva para 5 SSs (UNICAMP)

Tal como ocorreu na análise I com este módulo, a taxa de transferência das ligações UGS demoraram cerca de 100 segundos a estabilizar. A partir desse momento cada SS com aplicações UGS apresenta um nível constante da taxa de transferência. Neste caso as ligações apresentam grandezas na ordem dos 65kbps. As variações entre os mínimos e máximos de cada ligação rondam 1kbps.

O tráfego efectuado pela SS5 referente à classe BE é mostrado na Figura 77.

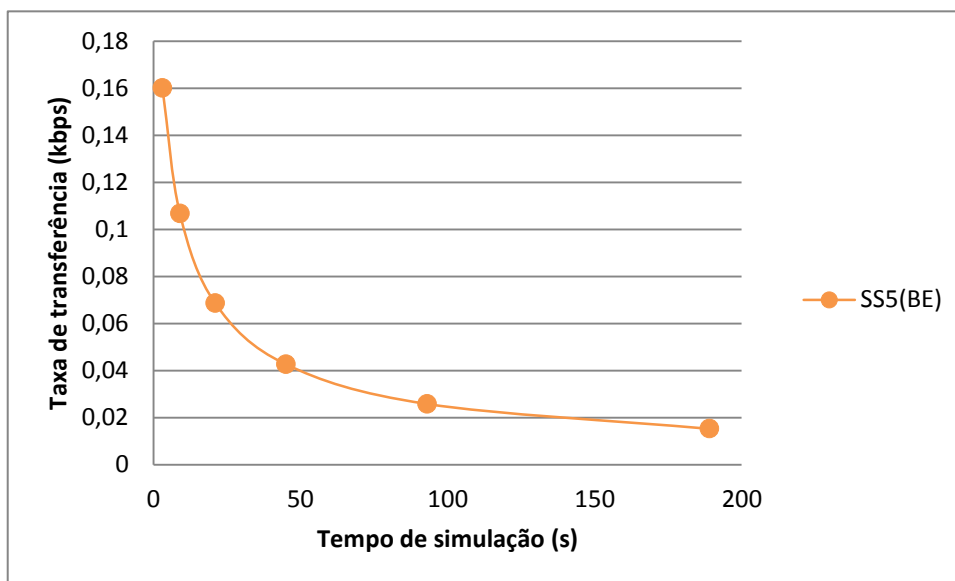


Figura 77 - Taxa de transferência efectiva SS5 (BE) – UNICAMP

Mais uma vez fica demonstrado que, em concorrência com tráfegos de prioridade superior este módulo é incapaz de proporcionar à classe BE as condições necessárias para que as aplicações possam ser executadas. O nível de taxa de transferência vai decrescendo ao longo do tempo de simulação.

ATRASOS

A Figura 78 mostra os atrasos médios medidos para as SSs que apresentam o tráfego UGS.

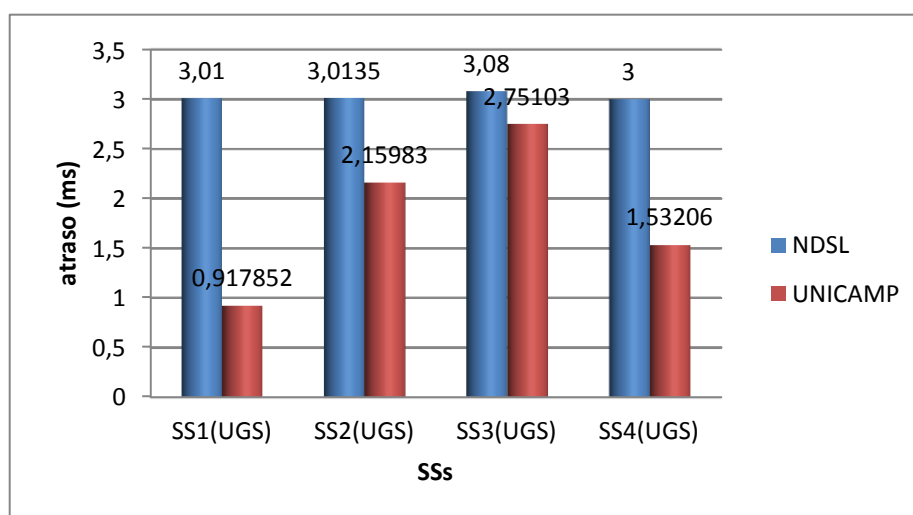


Figura 78 - Atrasos referentes ao 1º cenário, análise II

Como é possível verificar os valores de atraso de ambos os módulos são inferiores aos parâmetros necessários à classe de QoS UGS, 20ms (Tabela 10). Tanto o módulo NDSL

como o UNICAMP apresenta valores muito inferiores aos exigidos. Os valores encontrados no módulo implementado pela UNICAMP são inferiores aos encontrados no outro módulo. No caso do NDSL os atrasos médios de cada SS são de 3ms enquanto que no módulo da UNICAMP os atrasos médios são de 0.9ms, 2.15ms, 2.75ms, 1.53ms referentes às SSs 1, 2, 3 e 4 respectivamente.

COMPARAÇÃO DO COMPORTAMENTO DOS MÓDULOS PARA A 2ª ANÁLISE DO 1º CENÁRIO

Verificou-se que ambas as classes são capazes de lidar com várias ligações UGS em simultâneo, apresentando taxas de transferência efectivas constantes ao longo da simulação. Existe uma maior diferença das taxas entre cada SS no módulo UNICAMP. Os atrasos encontrados em cada SS, de ambos os módulos, demonstram o bom funcionamento dos módulos a este nível, apresentando, uma vez mais, valores bastantes inferiores aos requeridos. O módulo NDSL apresenta atrasos um pouco superiores aos encontrados no módulo UNICAMP.

6.4. 2º CENÁRIO DE SIMULAÇÃO – ESTUDO DO COMPORTAMENTO DO ESCALONADOR PARA CLASSES DE ALTA PRIORIDADE

No segundo cenário de simulação pretende-se testar o comportamento do escalonador nas classes de maior prioridade. As análises serão efectuadas de modo a verificar se os módulos são capazes de alocar os mesmos valores de largura de banda independentemente do número de SSs presente na rede, analisando-se também, o comportamento do escalonador para o caso de um elevado número de SSs. Os parâmetros de configuração das classes de maior prioridade, utilizadas nestas experiências, são os mesmos utilizados para o primeiro cenário de simulação. Nas simulações serão obtidos os valores médios da taxa de transferência efectiva e dos atrasos em função do número de SSs introduzidas na rede.

6.4.1. ANÁLISE I – ESTUDO DO COMPORTAMENTO DA CLASSE UGS

O objectivo da primeira análise é testar o comportamento do escalonador na classe de maior prioridade, aumentando significativamente o número de SSs, que possuem aplicações referentes ao tráfego UGS. A Figura 79 demonstra a topologia de rede utilizada nesta análise.

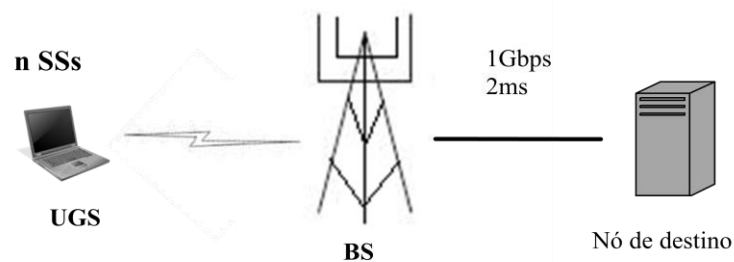


Figura 79 - Estrutura da rede – 2º cenário (análise I)

A rede presente na análise I é composta por 1 BS com SSs distribuídas uniformemente à sua volta. A BS está também ligada por cabo a um nó de destino.

RESULTADOS OBTIDOS

A Figura 80 ilustra a taxa de transferência efectiva de uma ligação UGS em função do aumento do número de SSs na rede. Cada SS possui uma ligação UGS com a BS.

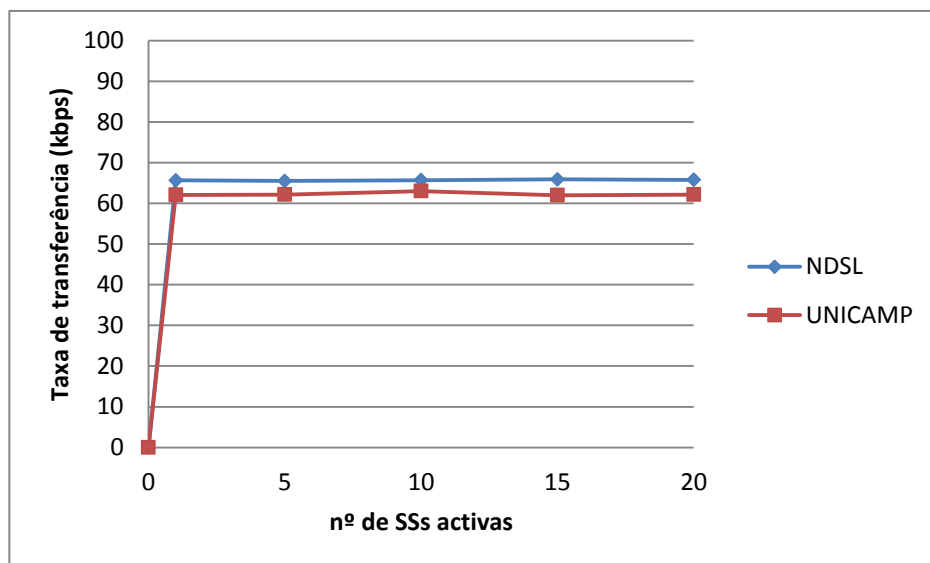


Figura 80 - Taxa de transferência de uma ligação UGS

É possível verificar que até às 20 SSs testadas os escalonadores, de ambos os módulos, são capazes de proporcionar, à classe UGS, taxas de transferência efectiva constantes, independentemente do número de SSs presentes na rede. Isto significa que a BS é capaz de alocar a quantidade necessária de *slots* para o bom funcionamento do tráfego UGS. Como se pode verificar no gráfico apresentado, mesmo com 20 SSs activas, a taxa de transferência das ligações UGS não sofre qualquer diminuição, demonstrando assim, que,

ambos os módulos apresentam resultados positivos no tratamento da classe de maior prioridade.

ATRASOS

A Figura 81 reporta-se aos atrasos médios encontrados, em ambos os módulos, para a classe UGS.

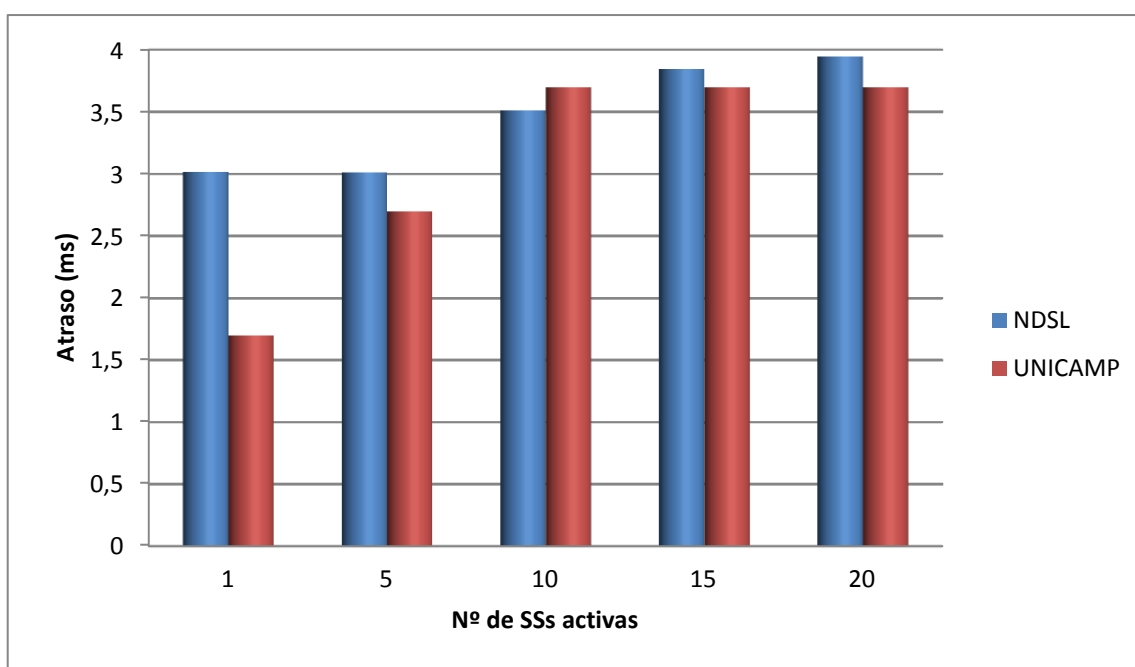


Figura 81 - Atrasos médios da classe UGS

Os dois módulos estudados nesta análise apresentam resultados indicadores do bom funcionamento do escalonador referente à classe UGS. Ambos os módulos exibem atrasos muito inferiores ao exigido para esta classe (20ms). Verifica-se que os atrasos médios encontrados no módulo UNICAMP tendem a estabilizar a partir de 10 SSs presentes na rede. É possível verificar que, mesmo com 20 SSs activas na rede, os atrasos encontrados para este tipo de tráfego mantêm-se ligeiramente acima dos 3,5ms.

6.4.2. ANÁLISE II – ESTUDO DO COMPORTAMENTO DA CLASSE RTPS

O objectivo dos testes realizados nesta secção é verificar o comportamento dos escalonadores dos módulos da NDSL e da UNICAMP quando submetidos a tráfego que requer os valores mais elevados de taxa de transferência, o tráfego rtPS. Pretende-se aferir se, com o aumento do número de ligações rtPS, a BS de cada módulo é capaz de alocar de uma forma justa a largura de banda a cada ligação. Nesta análise também são comparados

os atrasos médios medidos. A Figura 82 demonstra a topologia de rede utilizada nesta análise.

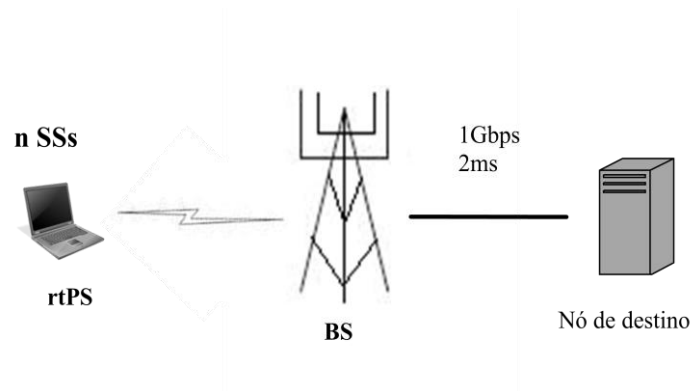


Figura 82 - Estrutura de rede – 2º cenário (análise II)

A rede presente na análise II é composta por 1 BS com SSs distribuídas uniformemente à sua volta. Cada SS apresenta uma ligação rtPS com a BS. A BS está também ligada por cabo a um nó de destino.

RESULTADOS OBTIDOS

A Figura 83 representa a taxa de transferência efectiva de uma ligação rtPS, em função do número de SSs na rede. Cada SS possui uma ligação rtPS com a BS.

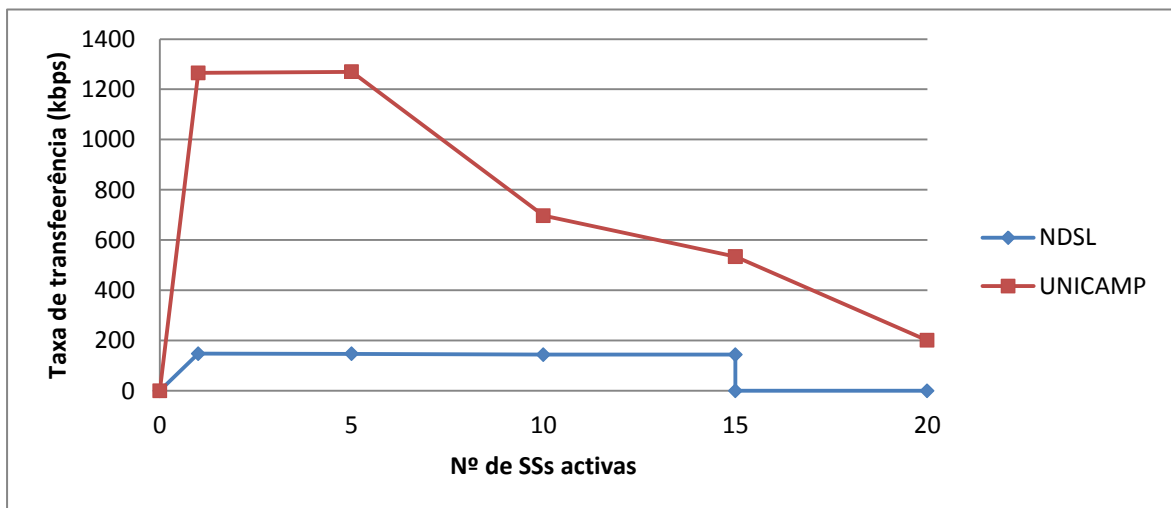


Figura 83 - Taxa de transferência efectiva – análise II

O módulo implementado pela UNICAMP apresenta resultados que indicam que o escalonador actua de forma justa relativamente à alocação da largura de banda para a classe rtPS. Com o aumento de SSs com aplicações pertencentes à classe rtPS, o escalonador vai ajustando a disponibilização da largura de banda a cada SS. Verifica-se

que, só a partir das 5 SSs é que o escalonador necessita de diminuir a largura de banda entregue a cada SS. Este módulo apresenta valores muito superiores aos encontrados no módulo NDSL demonstrando um melhor desempenho neste serviço.

No módulo NDSL é possível observar que até estarem 15 SSs na rede, o escalonador aloca a mesma quantidade de largura de banda a cada SS, 180kbps. A partir das 15 SSs, o módulo NDSL atinge a saturação, sendo incapaz de concluir o processo de simulação. Com isto, é possível concluir que este módulo é incapaz de efectuar simulações com redes que necessitem de altas taxas de transferência.

ATRASOS

A Figura 84 refere-se aos atrasos médios encontrados, em ambos os módulos, para a classe rtPS.

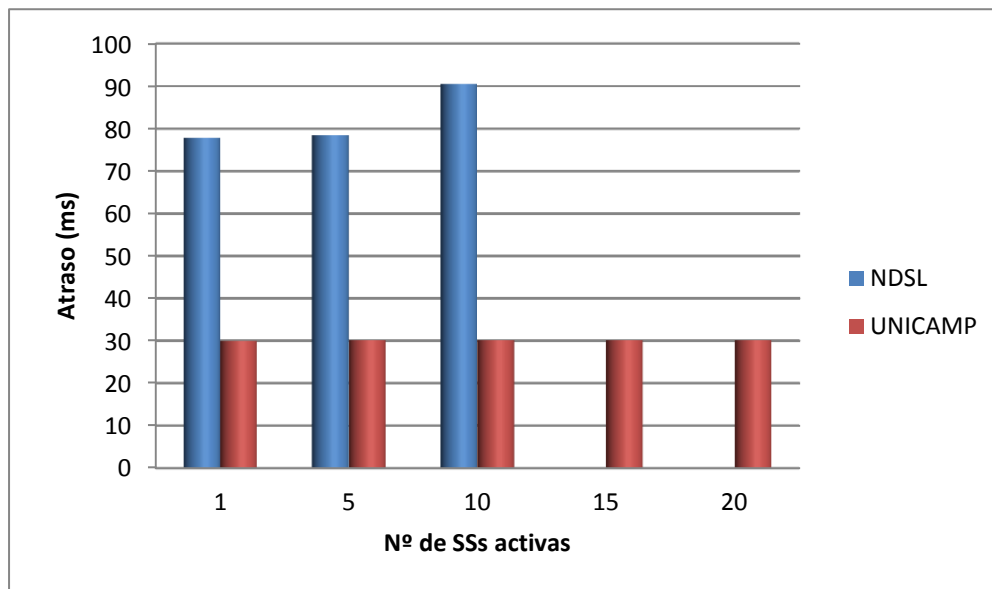


Figura 84 - Atrasos médios da classe rtPS

É possível observar na Figura 84 que os atrasos encontrados no módulo UNICAMP apresentam valores de atraso menores do que os exigidos a estas classes de serviços (60ms). Com o aumento do número de SSs na rede o atraso médio encontrado não sofre alteração mantendo-se constante, 30ms.

O módulo implementado pela NDSL é incapaz de apresentar resultados satisfatórios para esta classe de QoS. Apresenta atrasos superiores aos requeridos, sendo impossível efectuar simulações superiores a 15 SSs presentes na rede.

6.5. 3º CENÁRIO DE SIMULAÇÃO – ESTUDO DO COMPORTAMENTO DO ESCALONADOR PARA A CLASSE DE MENOR PRIORIDADE (BE)

O terceiro cenário tem o objectivo de estudar o processo de alocação de largura de banda entre ligações BE. Apesar do WIMAX suportar diferentes classes de serviço, é antecipado que o maior número de ligações efectuadas em simultâneo por diversas SS usará a classe BE. Assumindo este facto, é importante assegurar que os recursos de largura de banda disponível são partilhados de uma forma justa entre as ligações BE, independentemente do seu número e do facto deste serviço não possuir quaisquer requisitos de QoS.

Nesta análise serão considerados para o estudo os três módulos presentes neste trabalho, NDSL, UNICAMP e NIST.

Neste cenário cada SS estabelece uma ligação *uplink* com a BS, contendo uma aplicação FTP.

O tempo de simulação é de 100 segundos e foram executadas 5 simulações para a obtenção dos resultados.

6.5.1. ANÁLISE I – ESTUDO DE 1 SS COM TRÁFEGO BE

A Figura 85 representa a estrutura de rede efectuada na primeira análise do 3º cenário. Neste caso, a rede apresenta apenas uma SS e uma BS com uma ligação BE. A BS está também ligada por cabo a um nó de destino.

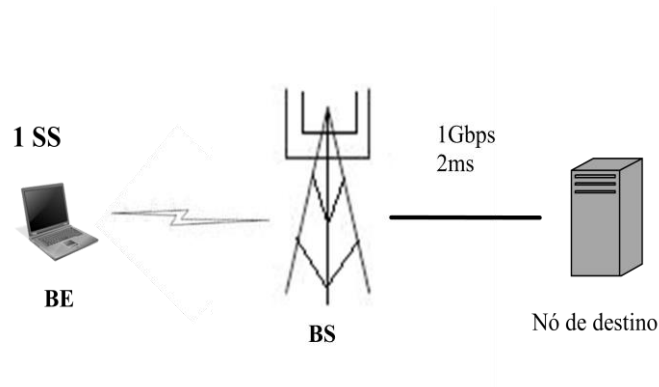


Figura 85 - Estrutura de rede – 3º cenário (análise I)

Esta análise tem como objectivo testar qual a taxa de transferência efectiva que a BS de cada módulo é capaz de oferecer à SS para o tráfego BE. A Figura 86 demonstra os resultados obtidos.

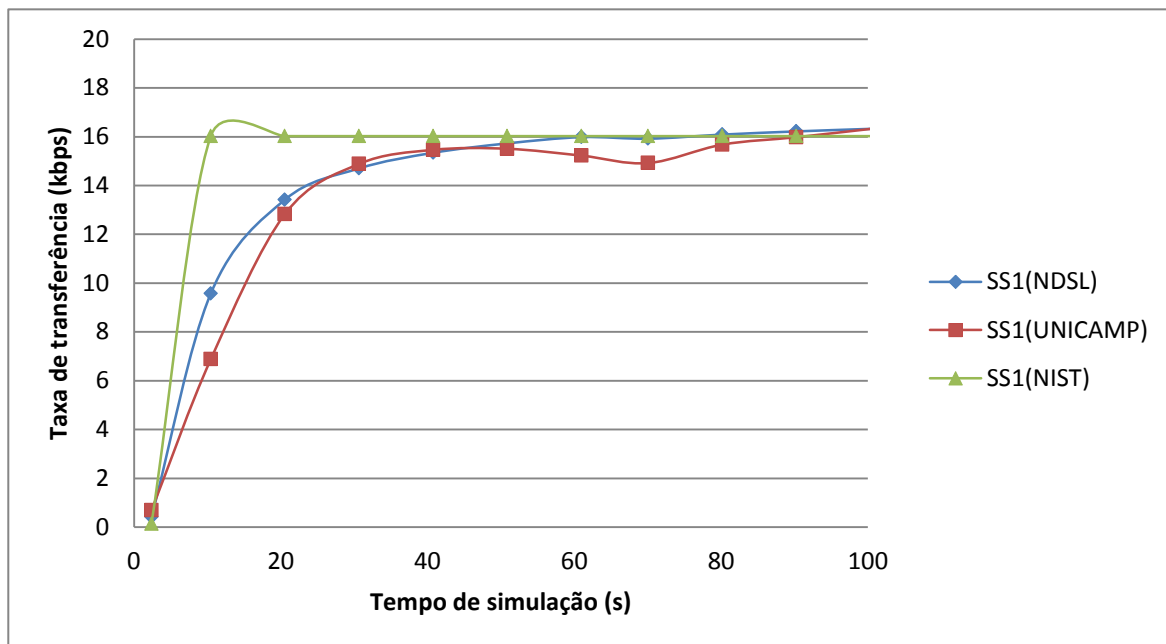


Figura 86 - Taxa de transferência efetiva para uma SS (BE)

É importante constatar que os módulos NDSL e UNICAMP limitam as aplicações BE para que a sua taxa de transferência não exceda um limite próximo de 16 kbps. Apesar de, neste cenário, não existir concorrência com outras classes de maior prioridade, cada ligação BE, tem disponíveis 16 kbps para as suas transmissões *uplink*. Assim, é possível concluir que, tanto o módulo implementado pela NDSL como o módulo implementado pela UNICAMP não proporcionam um tratamento correcto da classe de prioridade mais baixa (BE), onde, num cenário em que este tráfego é o único tráfego presente na rede, não deveria haver qualquer limitação ao nível da sua largura de banda.

Numa tentativa de igualar o sucedido com os módulos NDSL e UNICAMP, foi imposto que a taxa de transferência máxima da aplicação FTP, no módulo implementado pela NIST, seria de 16kbps. Com isto, pretendeu-se efectuar uma comparação justa entre os três módulos estudados. Neste contexto pode verificar-se que a resposta do módulo da NIST é a mais rápida.

6.5.2. ANÁLISE II – ESTUDO DA CONCORRÊNCIA ENTRE 15 SSS COM TRÁFEGO BE

A Figura 87 traduz a estrutura da rede para a segunda análise de simulação. A rede inclui uma BS e quinze SSS que utilizam ligações BE. A BS apresenta uma ligação por cabo a um nó de destino.

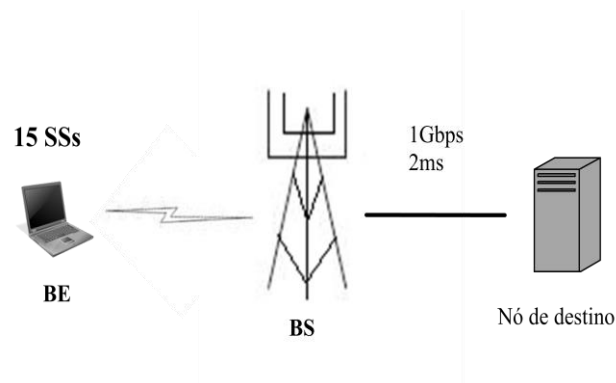


Figura 87 - Estrutura da rede – 3º cenário (análise II)

Cada SS estabelece uma ligação *uplink* com a BS, simulando uma aplicação FTP que envia dados. O objectivo deste caso de estudo é verificar se a BS é capaz de alocar de forma justa a largura de banda a 15 SSs que entram na rede em simultâneo, no início da simulação, e terminam as suas aplicações no fim da simulação.

RESULTADOS OBTIDOS – MÓDULO NDSL

A Figura 88 seguinte representa a taxa de transferência efectiva de cada SS.

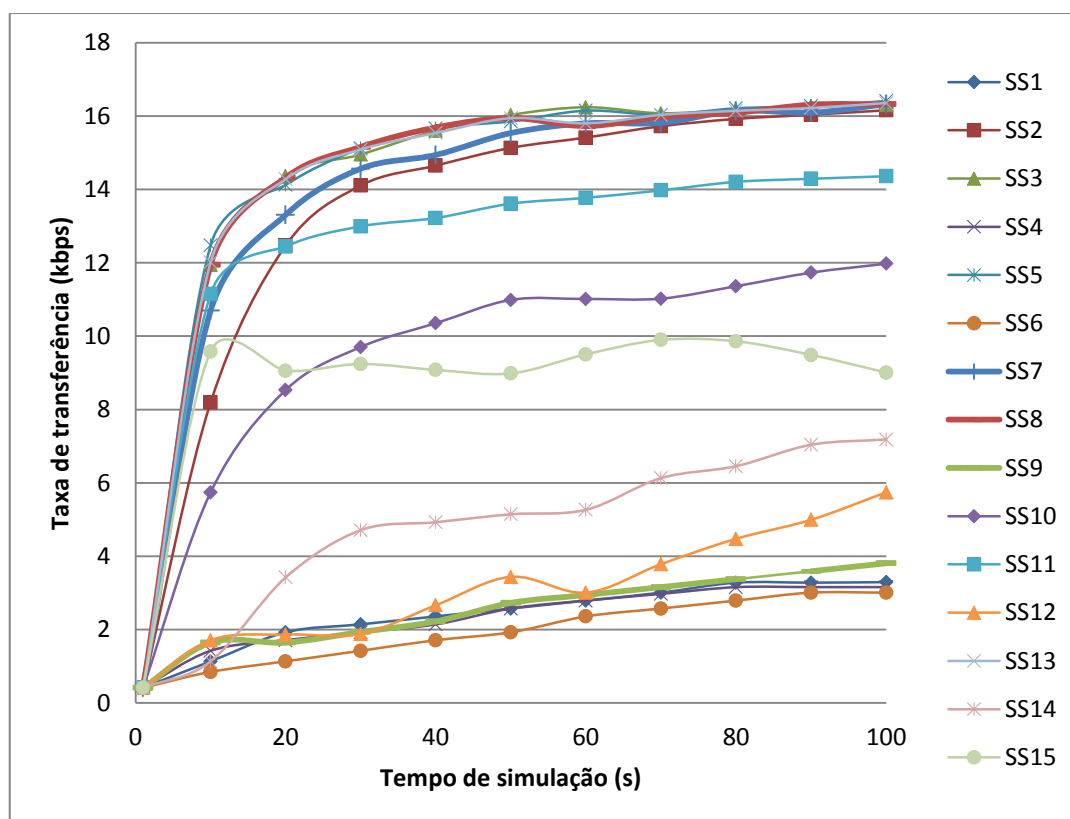


Figura 88 - Taxa de transferência para 15 SSs b) (NDSL)

Analisando o gráfico é possível verificar que a BS é incapaz de alocar a largura de banda necessária para que esta seja distribuída por todas as SSs de uma forma justa. É possível constatar que a taxa de transferência é atribuída de uma forma completamente aleatória a cada SS, demonstrando que as SSs que requisitaram primeiro a largura de banda são aquelas que apresentam uma maior taxa de transferência. É importante referir que os pedidos de largura de banda também são feitos de forma aleatória uma vez que as SSs entram na rede ao mesmo tempo. Às últimas SSs a fazerem a mesma requisição, a BS aloca a largura de banda que nesse momento terá à sua disposição para este tipo de tráfego.

RESULTADOS OBTIDOS – MÓDULO UNICAMP

A Figura 89 apresenta a taxa de transferência efectiva para cada uma das SSs ao longo do tempo de simulação.

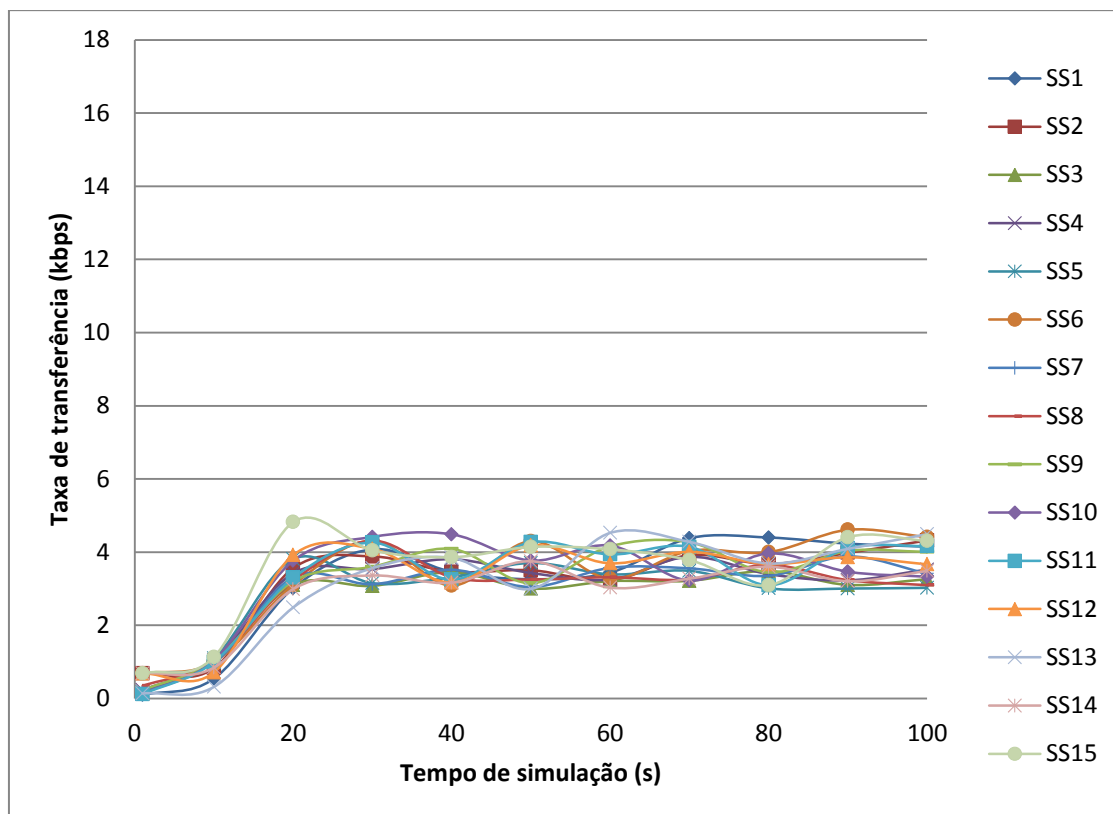


Figura 89 - Taxa de transferência efectiva para 15 SSs b) (UNICAMP)

O módulo da UNICAMP é capaz de alocar de forma justa a largura de banda disponível para a classe de QoS de menor prioridade (BE). Verifica-se que a taxa de transferência efectiva para cada SS atinge valores entre os 3 e os 4 kbps. Note-se que esta taxa de transferência é bastante baixa demonstrando que este módulo, no que respeita à classe BE,

apresenta um limite de largura de banda muito baixo, o que não se justifica quando a classe BE é a única presente na rede.

RESULTADOS OBTIDOS – MÓDULO NIST

A Figura 90 demonstra os resultados obtidos na simulação efectuada com o módulo implementado pela NIST.

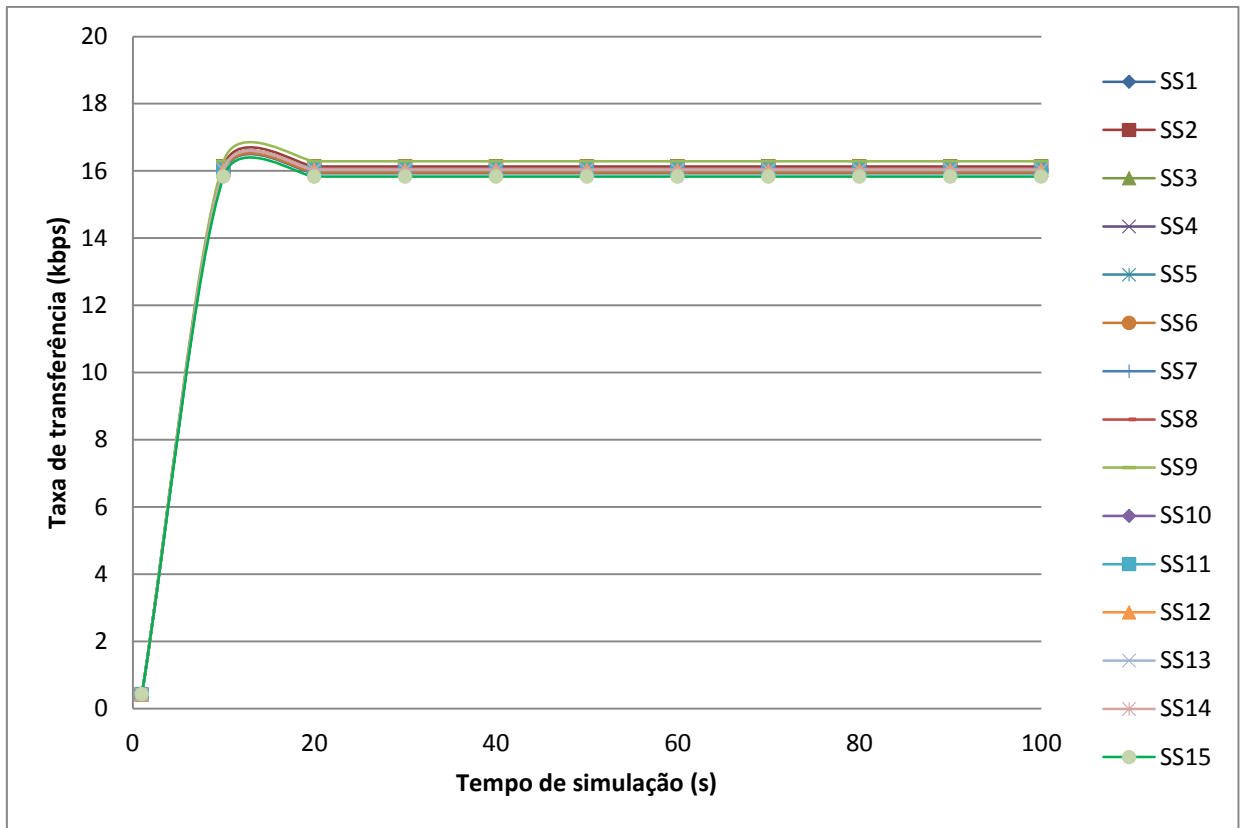


Figura 90 - Taxa de transferência efectiva para 15 SSs – módulo NIST

Analisando o gráfico é possível concluir que a largura de banda é distribuída uniformemente por todas as SSs presentes na rede. Verifica-se que este módulo, para baixas taxas de transferências (16kbps), é capaz de atribuir a quantidade de *slots* necessária a cada SS para que esta possa utilizar a taxa de transferência máxima imposta nas configurações da simulação.

6.5.3. ANÁLISE III – ESTUDO DA CONCORRÊNCIA ENTRE 15 SSS COM TRÁFEGO BE VARIANDO O Nº DE SSS ACTIVAS

A estrutura de rede nesta análise é idêntica à apresentada na análise II. Para efectuar o estudo da alocação de largura de banda com diferentes números de fluxos activos, foram

ligadas ou desligadas algumas das aplicações FTP ao longo da simulação. Se apenas algumas aplicações estiverem activas, o escalonador na BS deverá ser capaz de alocar mais *slots* para as SSs que estão activas o que deverá incrementar as suas taxas de transmissão. Consequentemente, se várias aplicações estiverem activas, a BS deverá alocar os recursos de uma forma equivalente entre as SSs. Como resultado, cada ligação possuirá apenas alguns *slots*.

Tabela 12 - Aplicações activas

SS	Tempo de simulação (s)			
	0-25	25-50	50-75	75-100
SS1 – SS3	√	√	√	
SS4 – SS5		√	√	
SS6 – SS13			√	
SS14 – SS15			√	√

A Tabela 12 expõe os intervalos de tempo em que cada SS está activa. No início apenas três estão activas (SS1-SS3). Ao fim de 25 segundos entram em actividade mais duas SSs (SS4, SS5). Dos 50 aos 75 segundos entram em acção mais 8 SSs (SS6-SS13), ficando nos últimos 25 segundos da simulação apenas duas SSs em actividade (SS14,SS15). Convém mencionar que todas as SSs entram na rede no início da simulação variando o tempo em que cada aplicação fica activa. Com isto, pretende-se que o escalonamento realizado pela BS seja feito através dos pedidos de cada SS e não pelo facto de cada SS ter entrado na rede.

RESULTADOS OBTIDOS – MÓDULO NDSL

A Figura 91 representa a taxa de transferência efectiva para cada SS do módulo NDSL.

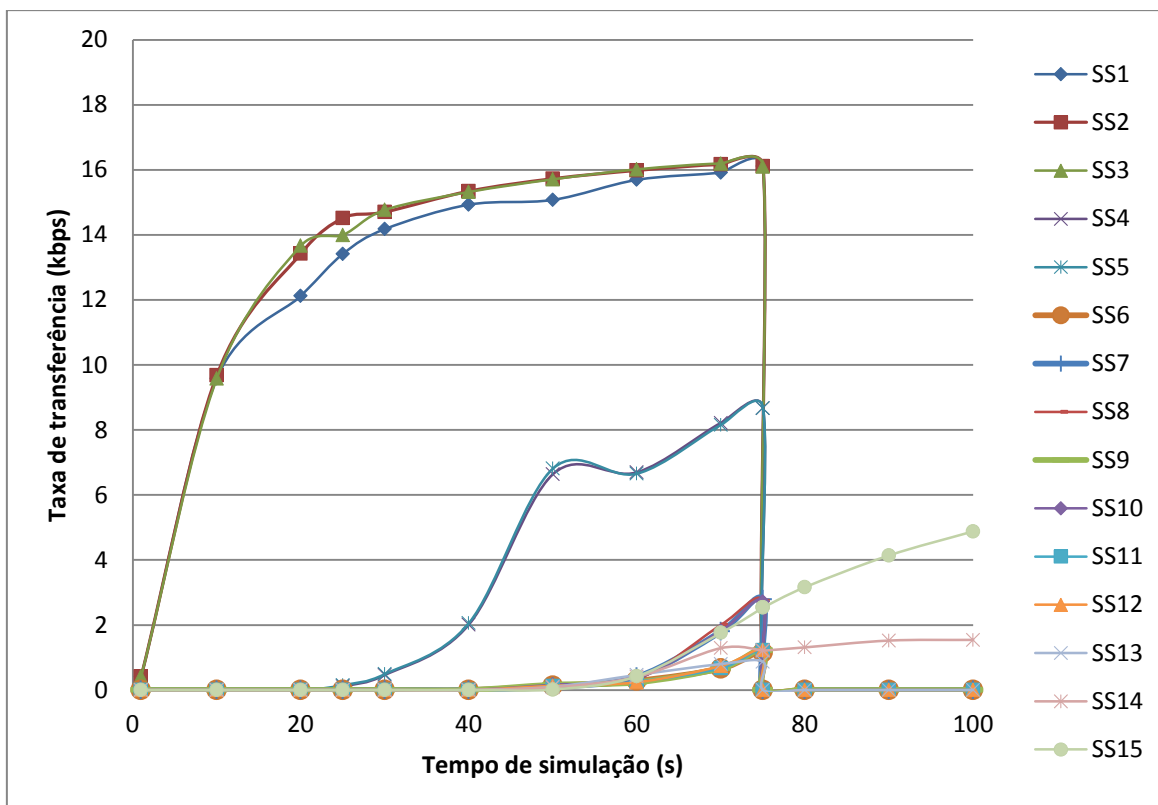


Figura 91 - Taxa de transferência para 15 SSs – 3º cenário (análise III) (NDSL)

Como é possível constatar através do gráfico acima representado, o módulo implementado pela NDSL é incapaz de efectuar uma alocação justa da largura da banda para cada tráfego emitido pelas diferentes SSs. Verifica-se que as primeiras três SSs ao entrar em actividade são as que apresentam uma taxa de transferência efectiva maior. Ao fim dos primeiros 25 segundos verifica-se a entrada de mais duas aplicações anexadas às SS4 e SS5. Estando parte da largura de banda já reservada para as aplicações que se encontram activas, a taxa de transferência que as novas aplicações adquirem são bastantes inferiores. Aos 50 segundos de simulação entram em actividade 8 SSs que se juntaram às restantes já em funcionamento. Mais uma vez, estas novas SSs apresentam valores bastante inferiores. Apesar de, aos 75 segundos, todas as SSs que estavam activas terminarem a sua aplicação, o atraso que tráfego destas apresenta não permite que as duas novas aplicações anexadas às SSs 14 e 15, que acabaram de entrar na rede, possuam os níveis de taxa de transferência que, teoricamente, atingiriam se estivessem sós na rede.

RESULTADOS OBTIDOS – MÓDULO UNICAMP

A Figura 92 representa a taxa de transferência efectiva de cada uma das 15 SSs em função do tempo de simulação. Neste caso a simulação foi efectuada no módulo implementado pela UNICAMP.

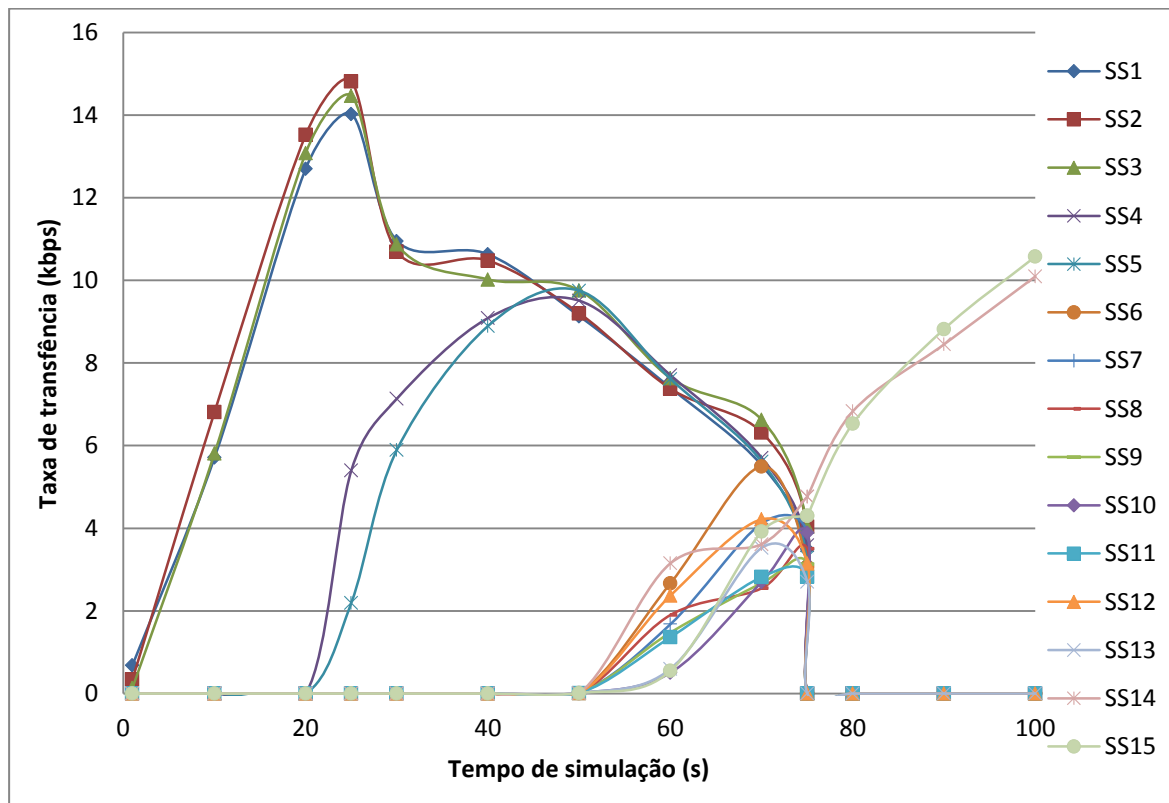


Figura 92 - Taxa de transferência para 15 SSs (UNICAMP)

Analisando o gráfico obtido verifica-se que o escalonador presente na BS é capaz de alocar, de forma justa, a largura de banda entre as aplicações activas anexadas a cada SS. Ao longo do tempo de simulação, a largura de banda atribuída a cada SS vai variando adaptando-se ao número de aplicações activas na rede. Nos primeiros 25 segundos as três SS com aplicações activas apresentam um nível de taxa de transferência na ordem dos 15kbps. Dos 25 aos 50 segundos, duas novas aplicações entram na rede. Neste intervalo pode-se constatar que as duas SSs apresentam valores similares de taxa de transferência. As SSs que já se encontravam na rede tendem a atingir o nível das novas SSs. O mesmo cenário acontece a partir dos 50 segundos quando estão presentes 13 aplicações na rede, baixando o valor das taxas de transferência de cada SS. No intervalo de tempo entre os 75 e os 100 segundos, as duas únicas aplicações presentes na rede (SS14 – SS15) apresentam um aumento das suas taxas de transferência efectiva, reflectindo o final das transmissões das restantes SSs.

RESULTADOS OBTIDOS – MÓDULO NIST

A Figura 93 representa a taxa de transferência em função do tempo para o módulo implementado pela NIST

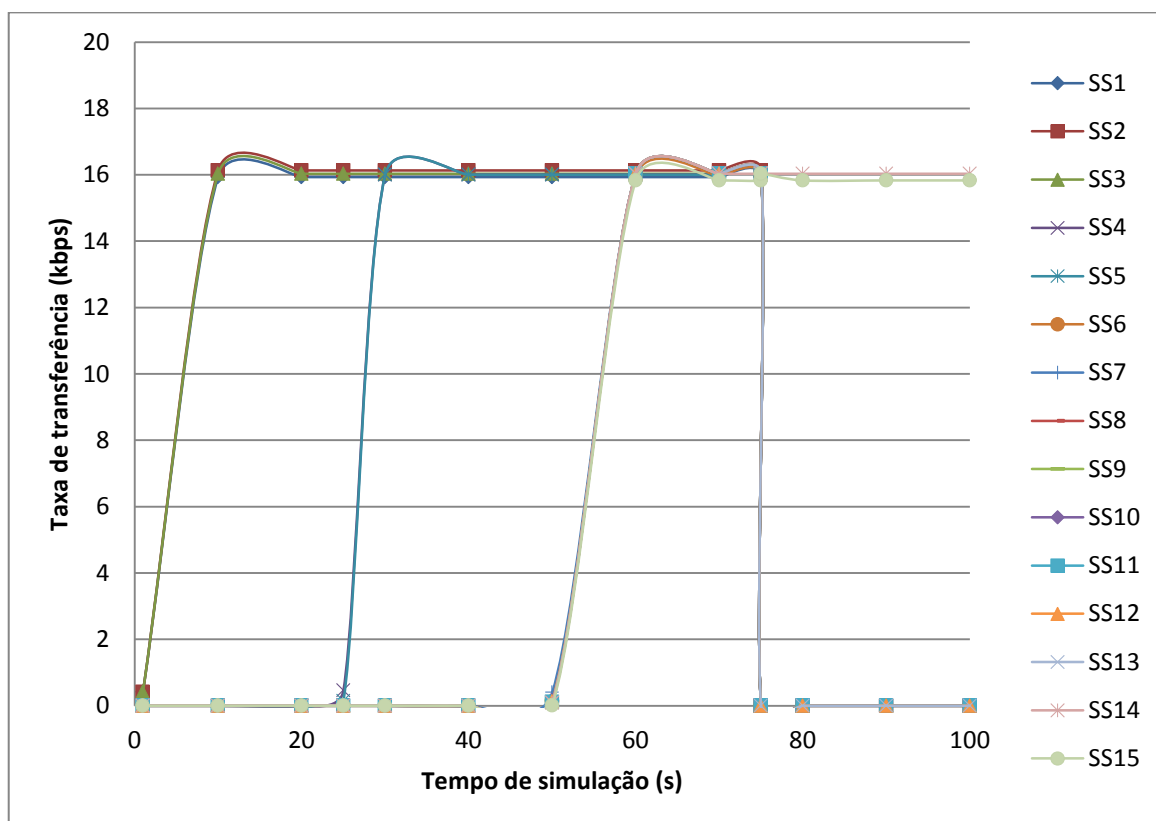


Figura 93 - Taxa de transferência para 15 SSs (NIST)

Analisando o gráfico verifica-se que o módulo da NIST aloca os *slots* de uma forma justa para todas as SSs. Neste caso constata-se que a taxa de transferência efectiva é sempre constante para todas as SSs, apresentando um valor igual a 16kbps. A entrada na rede das SSs não altera o valor da taxa de transferência das aplicações das SSs já presentes.

Verificou-se que, neste módulo, existe largura de banda suficiente para que cada ligação usufrua da largura de banda que necessita. Desta forma o escalonador não apresenta qualquer obstáculo às SSs na requisição da largura de banda.

Não havendo imposições limitadoras neste módulo, referente a este tipo de tráfego, a BS é capaz de alocar sempre os *slots* necessários, permitindo que os níveis de taxa de transferência efectiva não se alterem com a entrada ou saída de SSs na rede.

COMPARAÇÃO DO COMPORTAMENTO DOS MÓDULOS PARA O 3º CENÁRIO

Neste cenário verificou-se que o módulo criado pela NDSL é incapaz de efectuar a alocação da largura de banda para cada SS de uma forma justa. Os módulos implementados pela UNICAMP e pela NIST apresentam resultados que indicam o bom funcionamento do escalonador para situações de concorrência de acesso à largura de banda, demonstrando serem capazes de alocá-la de uma forma justa a cada ligação. É possível verificar que os módulos NDSL e UNICAMP apresentam um funcionamento da classe de menor prioridade de forma incorrecta, já que, quando esta classe é a única presente na rede deveria ser-lhe concedida a largura de banda máxima, por parte da BS. O módulo NIST, não tendo um mecanismo de escalonamento que faz a distinção entre as diversas classes, apresenta os melhores resultados para um cenário em que não foi testada a concorrência entre classes de serviço com outras de prioridades superior.

6.6. RESUMO

O sexto capítulo discute a parte da simulação da tese. Foram testados os módulos WIMAX de forma a efectuar uma análise das suas funcionalidades através de simulações na ferramenta ns-2.

Depois de serem replicados alguns cenários de simulação de forma a efectuar a comparação com os valores publicados pelos autores de cada módulo, foram produzidos três cenários de simulação na topologia PMP onde são feitas análises comparativas da eficiência de cada módulo, utilizando como termos de comparação a taxa de transferência efectiva e os atrasos médios encontrados.

O primeiro cenário apresenta um caso de múltiplos serviços. O objectivo deste cenário é testar se o escalonador de cada módulo, implementado na BS, é capaz de ter em consideração os diferentes tipos de classes de QoS, alocando os *slots* de acordo com as especificações requeridas por cada uma destas classes.

No segundo cenário de simulação pretende-se testar o comportamento do escalonador nas classes de maior prioridade. As análises serão efectuadas de modo a verificar se os módulos são capazes de alocar os mesmos valores de largura de banda independentemente do número de SSs presente na rede, analisando também, o comportamento do escalonador para o caso de um elevado número de SSs.

O terceiro cenário tem o objectivo de estudar como é feita a alocação de largura de banda entre ligações BE. É importante assegurar que os recursos de largura de banda disponíveis são partilhados de uma forma justa entre as ligações BE independentemente do seu número e do facto deste serviço não possuir quaisquer requisitos de QoS.

Os resultados encontrados nos três cenários demonstram que o módulo implementado pela UNICAMP apresenta resultados mais satisfatórios, quando comparado com o módulo implementado pela NDSL. Para o módulo criado pela NIST apenas foi possível realizar o terceiro cenário de simulação, já que este é incapaz de realizar diferenciação entre as diversas classes de QoS, tratando cada ligação como se pertence-se à classe BE.

7. CONCLUSÕES

Este trabalho apresenta um estudo da tecnologia WIMAX aprofundando a investigação na avaliação de três módulos implementados e disponíveis para simulação de redes WIMAX com a ferramenta ns-2.

Os três módulos investigados foram testados com a versão 2.32 da ferramenta ns-2. Nesta investigação foram motivos de análise os seus modos de operação, a geração de cenários em cada módulo e também os seus funcionamentos. Devido ao facto dos módulos não terem sido implementados para a versão do ns utilizada neste trabalho, foi necessário efectuar vários ajustamentos nos módulos da NDSL e da UNICAMP para que fosse possível a sua análise.

Depois de elaborado um estudo detalhado dos módulos, foram projectadas várias simulações para avaliar a implementação das suas principais funcionalidades, focando as experiências na área da gestão de QoS, bem como ao nível da alocação da largura de banda em cenários com ligações BE.

No capítulo da descrição dos módulos procurou-se caracterizar cada um dos três, tentando identificar-se as principais vantagens e inconvenientes. Verificou-se que o módulo da NIST é bastante mais completo que o módulo da NDSL e da UNICAMP, apresentando uma solução bastante mais agradável do ponto de vista do utilizador. A grande vantagem

dos módulos desenvolvidos pela UNICAMP e pela NDSL é o facto de estes já virem providos de mecanismos de controlo de classes de QoS.

Foram também reunidos e apresentados, neste estudo, os trabalhos científicos mais relevantes na área da gestão de QoS em redes WIMAX. Estes trabalhos não foram disponibilizados no domínio público.

Após certos ajustamentos efectuados aos três módulos, foram reproduzidas as simulações reportadas pelos autores dos diversos módulos, com o intuito de aferir a equivalência das configurações usadas e a tentativa de equivaler os resultados encontrados com os resultados obtidos nos artigos que acompanham cada módulo. Após se constatar a semelhança desses resultados foi possível passar para a simulação de novos cenários. A criação de cada cenário teve como objectivo realizar uma comparação entre os módulos, de forma a aferir qual o que apresenta a melhor performance.

Para o módulo criado pela UNICAMP, os resultados indicam que a BS é capaz de fornecer às SSs, praticamente todos os requisitos necessários correspondentes às especificidades de cada tipo de classe. Num primeiro cenário o comportamento do módulo, depois de um certo tempo de estabilização, foi o esperado. Tanto os atrasos como as taxas de transferência encontradas demonstraram estar entre os níveis requeridos pelas classes de QoS de alta prioridade. O único defeito encontrado refere-se ao tratamento das classes de baixa prioridade, que, sempre que concorrem com classes de prioridade superior, o escalonador não consegue alocar a largura de banda para o funcionamento deste tipo de ligações. Num segundo cenário, uma vez mais este módulo apresentou um bom funcionamento do escalonador. As ligações UGS e rtPS presentes evidenciaram atrasos e níveis de largura de banda dentro das limitações impostas para cada uma destas classes. Num último cenário estudado, onde se consideraram 15 SSs com ligações BE, foi possível verificar que este módulo é capaz de alocar de uma forma justa a largura de banda a cada SS, apresentando um ajustamento da taxa de transferência efectiva com a entrada de mais SSs na rede. A entrada de SSs na rede produz uma diminuição do nível dessa taxa, de forma a garantir que as SSs encontrassem todas as mesmas condições.

O módulo produzido pela NDSL apresenta resultados não tão satisfatórios quanto o módulo da UNICAMP. No primeiro cenário, a BS é capaz de atribuir os níveis de taxa de transferência necessários ao funcionamento das 5 classes de QoS, apesar de limitar a taxa

de transferência da classe rtPS. É possível verificar que o escalonador é capaz de atribuir às classes de menor prioridade (nrtPS e BE) alguma parte da largura de banda para que estas possam funcionar. Isto deve-se ao facto do sistema utilizar o método de escalonamento *weighted round-robin*, em que, no primeiro *round* de alocação de largura de banda, o escalonador atribui uma percentagem igual às 5 classes de QoS atribuindo a cada classe parte da largura de banda. Relativamente aos atrasos encontrados nas classes de alta prioridade é possível constatar que o módulo NDSL apenas é capaz de apresentar valores positivos para a classe UGS. As classes ertPS e rtPS apresentam valores acima dos requeridos. O segundo cenário demonstra que o escalonador deste módulo é capaz de alocar de uma forma justa a largura de banda necessária para as ligações UGS. Diferentemente do sucedido nestas ligações, quando são estudadas as ligações rtPS, que apresentam maiores níveis de taxa de transferência efectiva verifica-se que o módulo apenas suporta uma arquitectura de rede composta até 15 SSs com este tipo de aplicações. A partir deste valor, a simulação é abortada demonstrando um mau funcionamento do módulo quando este é sujeito a várias ligações com altos níveis de taxa de transferência. No terceiro cenário, este módulo é incapaz de alocar de forma justa a largura de banda a cada ligação. Nas análises efectuadas neste cenário é possível constatar que as primeiras SSs a entrarem na rede possuem os valores mais altos de taxa de transferência efectiva. Na análise III sempre que entram novas SSs na rede, como grande parte da largura de banda que o escalonador disponibiliza para esta classe já está a ser utilizada, estas ligações irão usufruir de uma menor largura de banda comparativamente com as restantes SS já activas.

Devido ao módulo implementado pela NIST não proporcionar o escalonamento de QoS, os resultados apresentados referem-se apenas ao terceiro cenário de simulação, um caso onde apenas existem ligações BE. Como a BS deste módulo, sem escalonador, é incapaz de fornecer às SSs diferentes tipos de tráfego, esta trata todas as SSs como se pertencessem à classe BE. Os resultados obtidos indicam que o escalonador é capaz de efectuar a alocação da largura de banda de uma forma justa. Para tentar igualar os outros dois módulos ao nível da máxima taxa de transferência efectiva utilizada nesta classe, foi posto um limite de 16kbps para cada ligação. Com isto verificou-se que a largura de banda presente neste módulo do WIMAX é capaz de fornecer a cada SS a largura de banda requisitada, não apresentando nenhuma diminuição do valor da taxa de transferência efectiva ao longo de todo o tempo de simulação. Assim, quando se efectuam simulações onde as aplicações não exijam requisitos de QoS, o módulo apresentado pela NIST, não limitando cada tráfego,

apresenta o resultado esperado para este tipo de cenário, demonstrando o mau funcionamento dos outros dois módulos.

Efectuando uma comparação entre os módulos que apresentam um escalonador capaz de gerir as diferentes classes de QoS é possível afirmar que o módulo criado pela UNICAMP apresenta resultados mais atractivos, mostrando uma maior qualidade na gestão das classes de maior prioridade. O entrave deste módulo, refere-se ao tratamento das classes de menor prioridade, onde não existem requisitos mínimos a atingir. O módulo NDSL apresenta melhores resultados neste caso devido à sua política de escalonamento.

O principal aspecto a ter em consideração quando se compara o módulo implementado pela NIST com os outros dois, é o facto deste ser capaz de efectuar simulações de cenários com mobilidade dos nós, apresentando ainda *handover* de BSs.

A partir deste trabalho é possível verificar que são possíveis inúmeros trabalhos no âmbito da tecnologia WIMAX. Este trabalho conclui que, ao nível de mecanismos de controlo de QoS existe um caminho a percorrer onde mais estudos sobre o tema serão com certeza apresentados. O entrave para que possam ser efectuadas comparações contundentes e conclusivas nesta área deve-se ao facto dos autores destas propostas de gestão de QoS não disponibilizarem as suas descobertas. Com isto é importante agradecer e salientar o esforço das equipas de investigadores que disponibilizaram os seus módulos para estudo. Esta disponibilização significa uma grande contribuição para o melhoramento desta tecnologia ainda imergente.

As simulações nestes três módulos e apresentadas neste trabalho são para topologias PMP referentes ao acesso fixo *wireless*. Até à data ainda não foram publicados estudos para o escalonamento de QoS respeitantes aos acessos móveis. Um trabalho interessante a ser realizado no futuro poderá ser o desenvolvimento de um mecanismo de escalonamento capaz de efectuar o controlo das classes de QoS em acessos móveis, poderia ser implementado a partir de um dos módulos já disponíveis, por exemplo o da NIST.

Um outro trabalho futuro muito interessante e audaz a ser desenvolvido é a criação de um módulo para a provisão de classes de QoS. Com isto, seria feito o esforço necessário para corrigir as lacunas encontradas nos módulos já disponibilizados.

Referências Documentais

- [1] ABICHAR Zakhia; PENG Yanlin.; CHANG J. Morris – *WiMAX: The Emergence of Wireless Broadband*. IT Pro, 2006.
- [2] CASTRO-PENA Rafael – *WIMAX Networks*. Paris Academy of Computer Science, SUPINFO Projects, 2005.
- [3] IEEE Std 802.11. Wireless Local Area Networks. Webpage, February 2006.
- [4] Wi-Fi Alliance. Wi-Fi Official Homepage. Webpage, February 2006.
- [5] WIMAX Forum – *Mobile WiMAX – Part I: A Technical Overview and Performance Evaluation*. Agosto, 2006.
- [6] IEEE Std 802.16-2004 – *Part 16 – Air Interface for Fixed Broadband Wireless Access Systems*. Technical report, June 2004. IEEE Standard for Local and metropolitan area networks.
- [7] IEEE Std 802.16e/D9 – *Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems - Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands*. Technical report, June 2005.
- [8] WIMAX Forum – *Fixed, nomadic, portable and mobile applications for 802.16-2004 and 802.16e WiMAX networks*. Novembro, 2005.
- [9] DIAS Gustavo – *IEEE 802.16 – WIMAX*. COS762 , 2005.
- [10] Lutz Fielbrandt. *WiMAX. Demystified*, Setembro, 2005. Funkschau Workshop WLAN & WiMAX.
- [11] ETSI. Welcome to ETSI! <http://www.etsi.org> .
- [12] NUAYMI Loutfi – *WiMAX: Technology for Broadband Wireless Access*. 2007.
- [13] WiMAX Forum – *WiMAX End-to-End Network Systems Architecture. Technical report*. Agosto, 2005.
- [14] SULLIVAN Declan; OBERL'E Julien; BOURKE Finola – *Fixed Wireless Access*. Sem data específica.
- [15] Howstuffworks.com – *How DSL Works*.
<http://computer.howstuffworks.com/dsl.htm>.
- [16] WIMAX Forum – *Can WIMAX Address Your Applications?*. Outubro, 2005.
- [17] UMTS World. *UMTS Handover*.
<http://www.umtsworld.com/technology/handover.htm>.
- [18] NAIR Govindan; CHOU Joey; MADEJSKI Tomasz; PERYCZ Krzysztof; PUTZOLU David; SYDIR Jerry – *IEEE 802.16 Medium Access Control and Service Provisioning*. Intel® Technology Journal, Volume 8, Issue 03, pag.213-228, 2004. ISSN 1535-864X.

- [19] SAYENKO Alexander, et al – *Ensuring the QoS Requirements in 802.16 Scheduling*. MSWiM'06, Outubro, 2006.
- [20] CHO Dong-Hoon, et al – *Performance Analysis of the IEEE 802.16 Wireless Metropolitan Area Network*. Proceedings of the First International Conference on Distributed Frameworks for Multimedia Applications (DFMA'05).
- [21] CHEN Jianfeng, et al – *An Integrated QoS Control Architecture for IEEE 802.16 Broadband Wireless Access Systems*. Sem data definida.
- [22] BELGHITH Aymen; NUAYMI Loutfi – *Design and Implementation of a QoS-included WiMAX Module for NS-2 Simulator*. Simutools 2008.
- [23] BELGHITH Aymen; NUAYMI Loutfi – *Scheduling algorithms for WIMAX*. Sem data definida.
- [24] SCHILLING Björn – *Qualitative Comparison of Network Simulation Tools – Modeling and Simulation of Computer Systems*. 2005.
- [25] LUCIO Gilberto; et al – *OPNET Modeler and Ns-2: Comparing the Accuracy Of Network Simulators for Packet-Level Analysis using a Network Testbed*. Sem data definida.
- [26] JAIN Richa – *GPRS Simulations using ns-Network Simulator*. Master's Thesis, 2001.
- [27] CHUNG Jae; CLAYPOOL Mark – *NS by Example*. <http://nile.wpi.edu/NS/> .
- [28] FALL Kevin; VARADHAN Kannan – *The ns Manual*. 2008.
- [29] HANBALI Ahmad – *Introduction NS2 Simulator*. 2007.
- [30] REZENDE José – *Simulador de redes ns*. GTA/COPPE/UFRJ.
- [31] MATTSSON Nils-Erik – *A DCCP module for ns-2*. Master's Thesis, 2004.
- [32] WELZL Michael – *Tutorial: The ns-2 Network Simulator*. Institute of Computer Science, University of Innsbruck.
- [33] CHENG Jenhui et al – *The Design and Implementation of WIMAX Module for ns-2 Simulator*. WNS2'06, Itália, 2006.
- [34] NDSL; CHENG Jenhui – *The Design & Implementation of WIMAX Module for ns-2 Simulator*. Sem data definida.
- [35] NDSL – *WiMAX for NS-2*. http://ndsl.csie.cgu.edu.tw/wimax_ns2.php .
- [36] NIST – *The Network Simulator NS-2 NIST add-on – IEEE 802.16 model (MAC + PHY)*. 2007.
- [37] FREITAG Juliana; FONSECA Nelson – *WIMAX MODULE FOR THE NS-2 SIMULATOR*. The 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2007.
- [38] FREITAG Juliana; FONSECA Nelson – *Um módulo para simulação de redes WIMAX no simulador ns-2*. 2008.
- [39] FREITAG Juliana; FONSECA Nelson – *Uplink Scheduling with Quality of Service in IEEE 802.16 Networks*. Sem data definida.

- [40] UNICAMP – *WIMAX module for the ns-2 simulator*.
http://www.lrc.ic.unicamp.br/wimax_ns2/ .
- [41] MANOS Papantonakos – *WIMAX – Evaluating VoIP using Network Simulator-2*. 2007.
- [42] KHAN Aadil – *QoS Support Multimedia in IEEE 802.16 Networks Simulation Results*. Sem data definida.
- [43] LIU Jen-Chu – *Qos Requirements: From applications Aspects*. MNET Lab. 2006.
- [44] DAWOOD Mohammed – *WiMAX & QoS*. ECOM 5301. 2007.